International Journal of Computer Applications Technology and Research
Volume 6–Issue 12, 467-472, 2017, ISSN:-2319–8656

A Novel Combined Approach for Daily Electric Load Forecasting Based on Artificial Neural Network and Modified Bat Algorithm

Eduardo Capra
The University of São Paulo
São Carlos, Brazil

Hugo Ribeiro
The University of São Paulo
São Carlos, Brazil

Abstract: In this paper a novel combined method based on Modified Bat Algorithm (MBA) and Neural Network algorithm has proposed in order to forecast the electric peak load power. In the proposed method, Bat Algorithm is employed as a popular optimization method and Artificial Neural Network is also utilized as a powerful mathematic method in mapping nonlinear relationship among model variables for the purpose of electric daily load prediction. Additionally, in order to improve the performance of the bat algorithm with regards to avoiding tapping into the local optimal and increasing the convergence speed, some modification has performed in bat algorithm which is called as SAMBA. Experimental results indicate that the proposed method has superiority performance in comparison with other traditional machine learning algorithms

Keywords: Short-Term Load Forecasting, Bat Algorithm, Artificial Neural Network, Intelligent Systems, Optimization Algorithm

1. INTRODUCTION

Electrical energy is considered as one of the mostly used forms of energy due to several benefits that can provide e.g. economical and clean type of energy [1], efficient transmission [2], easy distribution [3], availability [4], etc. Therefore, the demand for electrical energy has become an important issue in such a way that how this type of energy can be distributed and provided to the society efficiently [5-6]. Electrical load forecasting is defined as an intelligent process that predict required electrical power for short-term, medium-term, and long-term demand [7]. Designing the precise prediction model is essential in order to provide efficient and reliable electrical energy for all consumers. This demand has attracted the attention of scientific researcher to develop an accurate and efficient model for load forecasting [8-11].

Short-term load forecasting is categorized as a prediction of an hour up to a day ahead. Medium-term is considered for a seasonal prediction e.g. summer, winter, etc. additionally, it is used for fuel supplies scheduling for several days up to weeks. And Long-term Forecasting is generally used to make plan for growth of the required generating capacity and transmission that needs a prediction time from few months to few years [12].

In this paper our main focus is for short-term load forecasting by using machine learning algorithms. To this end, we combined both Bat algorithm as an optimization algorithm along with Artificial Neural Network (ANN) to provide an efficient and accurate model for the short-term time domain. In the following section, it will be demonstrated that the level of uncertainty which exists in the load forecasting problem is decreased noticeably. Recently, intelligent systems and machine learning algorithms have been widely used in different range of engineering practical problems. To name a few: using intelligent and expert method as system compensator and optimization method [13-14], modeling adaptive controller for industrial applications [15-16]. In particular, in the area of electric load forecasting, there are many methods available in the literature. In [17] authors proposed a new method that leverage the capability of chaos time series analysis to capture the electric load behavior for the short-term time horizon and then the nonlinear mapping of deterministic chaos is formulated by using perceptron algorithm. There are some other studies that employed time series analysis for load forecasting. [18-19]. In [20] a new method based on fuzzy regression analysis is introduced for short-term load forecasting error for 24 hours during the holidays. Authors believed that the average load forecasting error can be higher compared to the other days during week. In [21] an adaptive autoregressive moving-average model is introduced for electric load prediction and the superiority of their method is compared with the traditional Box-Jenkins transfer function approach. Recently, it is shown that Artificial Neural Network (ANN) has a potential capability of mapping nonlinear variables which are existed in the electric load data. Hence, it has caught the attraction of researcher in this area. In [22] authors present a hybrid model based on Radial Basis Function (RBF) neural network with adaptive neural fuzzy inference system. In this combined approach, the RBF network is firstly used to establish a model for load forecasting without considering the price change and then adaptive neural fuzzy inference system (ANFIS) is used by considering the recent real-time price change to adapt the results of load prediction which has attained by RBF previously. Emerging of evolutionary algorithms have gained great importance over the other existing techniques to adjust the different parameters of intelligent models due to their consistent and robust performance. [23-24]. There are many available evolutionary-based optimization techniques available in the literature for electric load forecasting namely Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Bat Algorithm (BA) [25-26], Crew Search Algorithm (CSA) [27], etc. In [28] the Artificial Neural Network is applied for hourly prediction of load forecasting and particle swarm optimization algorithm is utilized to tune the ANN weights and adjusting factor in the training phase. In [29] Genetic Algorithm is employed to find the most optimal parameters of Support Vector Regression (SVR) to enhance the accuracy and performance of SVR. Although GA is considered as powerful evolutionary-based optimization algorithm, the dependency of this method on its parameter can create a problem of trapping in the local optima. PSO can also face two similar shortcomings: dependency of initial values and trapping in local optima.

As it was mentioned previously, the proposed method made up of Artificial Neural Network (ANN) and modified Bat algorithm. The method follows these two steps: Firstly, the ANN is trained based on traditional training algorithm e.g. backpropagation and then the most suitable ANN architecture which includes number of input layers, number of hidden layers, number of output layer, number of neurons in each corresponding layer, adjusting and weighting factors. Secondly, a modified Bat optimization algorithm is employed to figure out the most optimized parameters for adjusting and weighting factors of ANN structure. Bat Algorithm can also have the limitations which we discussed earlier for GA and PSO (local optima trapping and premature convergence). In order to prevent algorithm to be trapped in local optima and increase convergence maturity, a new modification is introduced for BA. The main contribution of this paper is given as follows: 1) Finding optimal parameter of ANN based on modified Bat algorithm after the network being trained by classical methods. 2) applying the proposed method for prediction of electric load for the upcoming day.

2. Multi-Layer Feedforward Neural Network

Artificial Neural Network is considered as a popular and powerful machine learning method for classification and prediction which is inspired by biological system of human's brain. One of the mostly used application of this method is for forecasting. There are several distinctive features that make ANN as a powerful technique for forecasting (particular for electric load forecasting) e.g. mapping non-linear relations of variables. In general, multi-layer feedforward neural network consists of input layer, one or multi-level of hidden layer, and output layer. There are neurons in each layer and the connection of these layers are done by adjusting and weighting factors. Shown in figure 1 is an example of ANN with two levels hidden layer.

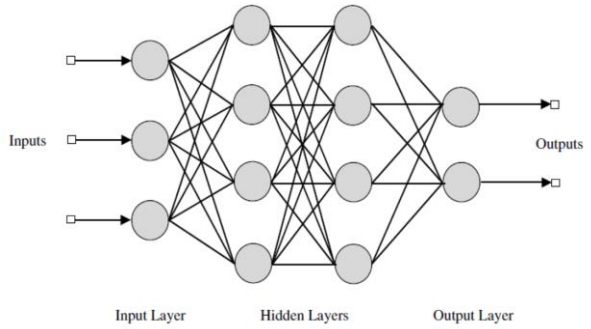


Figure 1. high-level structure of Artificial Neural Network

The determinations of ANN pattern include number of input layers, number of hidden layer, and number of output layer. The number of neurons which are used in input and output layer determine the number of input and output variables, respectively. There is a transfer function in ANN to map the input and output variables. In this paper, the sigmoid transfer function is chosen. Eq. 1 defines the output of each node in ANN.

$$y_i = f_1\left(\sum_{j=1}^{N_h}\left(w^2{}_{i,j}\, f_2\left(\sum_{k=1}^{N_i} w^1{}_{j,k} x_k + b^1{}_j\right) + b^2{}_i\right)\right) \quad (1)$$

Where:

f_1: input layer transfer function.

f_2: hidden layer transfer function.

b^1: hidden layer biasing metric.

b^2: output layer biasing metric.

$w^1{}_{i,j}$ weighting parameter to connect input node *(i)* to the node *(j)* in hidden layer.

$w^2{}_{i,j}$ weighting parameter to connect hidden layer node *(i)* to the node *(j)* in output layer.

3. Bat Algorithm

Bat Algorithm is inspired by the echolocation behavior of bats and considered as a new nature-inspired metaheuristic optimization algorithm [30]. This algorithm is inspired by this behavior of bats which they emit sound pulse and listen to the echo that is returning back from obstacles. Bat algorithm is formulated based on below for steps [31]:

1) Bats use echolocation behavior to know the distance of obstacles and figure out the difference between food and prey.

2) Each bat located in the position of X_i in the search space and flies with the velocity of V_i and emitting a particular pulse with the frequency and loudness of f_i and A_i respectively.

3) The loudness of A_i can be varied in many ways like by decreasing from a large value to a low value

4) The frequency f_i and rate r_i of each particular pulse is regulated automatically.

Having had a random fly, the location of each bat will be updated as follows:

$$\begin{aligned} \mathbf{V}_i^{new} &= \mathbf{V}_i^{old} + f_i(Gbest - X_i);\ i = 1,\ldots,N_{Bat} \\ X_i^{new} &= X_i^{old} + \mathbf{V}_i^{new} \quad ;\ i = 1,\ldots,N_{Bat} \\ f_i &= f_i^{\min} + \varphi_1(f_i^{\max} - f_i^{\min}) \quad ;\ i = 1,\ldots,N_{Bat} \end{aligned} \quad (2)$$

Where:

Gbest: the best bat from the objective perspective.

N_{Bat}: total number of bat population.

φ_1: random value in the range of [0,1].

$f_i^{\max}$ / $f_i^{\min}$: maximum/minimum frequency values of the i^{th} bat, respectively.

In order to receive the better random walking, the random number β is generated and if the new randomly generated number β is greater than r_i, the new solution is created as follows:

$$X_i^{new} = X_i^{old} + \varepsilon A_{mean}^{old};\ i = 1,\ldots,N_{Bat} \quad (3)$$

ε: random value in the interval of [-1,1]

A_{mean}^{old} : mean value of loudness of all bats.

If the randomly generated value β is less than r_i, new position is generated randomly. It should be noted that the following condition should be met in order to accept the new position.

$$[\beta < A_i] \& [f(X_i) < f(Gbest)] \quad (4)$$

Additionally, the loudness and rate of each pulse is updated as follows:

$$A_i^{new} = \alpha A_i^{old}$$
$$r_i^{Iter+1} = r_i^0 [1 - exp(-\gamma \times Iter)] \quad (5)$$

where α and γ are constant values and *Iter* is the number of the iteration during the optimization process.

3.1 Modified Bat Algorithm

As discussed earlier, in order to enhance the overall capability of Bat algorithm to prevent algorithm to be trapped in the local optima and premature convergence. The proposed modification includes two methods:

1. ***First modification:***

   In Eq. 5 the value of α will be updated during the optimization procedure by following formula:

   $$\alpha^{new} = (1/2Iter)^{1/Iter} \alpha^{old} \quad (6)$$

2. ***Second Modification:***

   This modification has introduced in order to increase the diversity of total bat population.

   $$X_i^{new} = \begin{cases} x_{i,j}^{old} & ; \varphi_1 < \varphi_2 \\ gbest_j & ; \varphi_1 \geq \varphi_2 \end{cases}$$
   $$X_i^{new} = [x_{i,1}^{new}, x_{i,2}^{new}, ..., x_{i,n}^{new}]$$
   $$Gbest = [gbest_1, gbest_2, ..., gbest_n] \quad (7)$$

   φ_2: random value between the range of [0,1].

   n: length of the control vector.

In the beginning of the procedure, each modification method has an equal probability ($Prob_\omega$ =0.5 & ω =1,2). Each bat selects the ω^{th} modification method by using roulette wheel mechanism (RWM). Hence, bats population is sorted out after each loop and the probability of each modification is updated as follows:

$$Prob_\omega = SR_\omega / \sum_{\omega=1}^{2} SR_\omega ; \omega = 1, ..., 2 \quad (8)$$

SR_ω: successful rate of ω^{th} modification method for generating optimal bats within the population. The criteria to be more optimal for each bat is defined such that if each new bat's objective function is less than the previous iteration. Having updated the probability of each modification, φ_3 is created as a random number in the range of [0,1]. The procedure of selecting the proper modification method ω^{th} will be chosen by using roulette RWM to create the next bat population generation.

4. Proposed Combined Method Based-on ANN and Modified Bat Algorithm:

In order to leverage the capability of the modified bat algorithm to adjust the ANN parameters, each member of the bat society $\overline{X}_i$ includes both weight and biasing factor of ANN which is defined in below equation:

$$\overline{X}_i = [w_{i,1}, ..., w_{i,Nw}, b_{i,1}, ..., b_{i,Nb}]_{(1,Nw+Nb)} \quad (9)$$

Bat society size is the same as the total number of all weighting and biasing factors of ANN. In order to demonstrate the accuracy and performance of the proposed method, two performance measurement criteria are introduced: relative error and Mean Absolute Percentage Error (MAPE) which are formulated as follows, respectively:

$$\sigma_i \% = \frac{|P_i - T_i|}{T_i} \times 100, i = 1, 2, ..., N_{es} \quad (10)$$

$$MAPE\% = \frac{1}{N_{es}} \sum_{i=1}^{N_{es}} \sigma_i \quad (11)$$

P_i: estimated value of the i^{th} data.
T_i actual value of the i^{th} data.
N_{es}: number of data that supposed to be forecasted.

Generally, the less MAPE value, the more accurate forecasting results will be attained. Therefore, the objective function of the proposed method should minimize MAPE value. The procedure of the proposed method is summarized as follows:

Step 1*:* Generating required parameters: All the input parameters such as number of the input neurons, number of output neurons, training algorithm, bat population size, the initial value, maximum and minimum value for loudness pulse of each bat should be initialized.

Step 2: Data normalization: In order to have a satisfactory results and structure for ANN, data normalization is required.

Step3: Weighting and biasing factors optimization: In this step the proposed method is employed to optimize both weight and biasing factors in ANN. The objective function for this purpose is to keep the MAPE values as minimum.

Step 4: Generating initial population for Bat algorithm: the bat population vector is described as follows:

$$\text{Bat\_}population = \begin{bmatrix} X_1 \\ X_2 \\ ... \\ X_{N_{Bat}} \end{bmatrix}_{N_{Bat} \times (N_w, N_b)} \quad (12)$$

N_{bat} is defined as the number of whole bats.

Step 5: Objective function measurement: In this step MAPE criterion will be assessed for all bats in population.

Step 6: Defining the best bat (*Gbest*): The best X_i is chosen as a bat with the least value of MAPE as objective function.

Step 7: Apply the proposed modification bat algorithm to enhance the population diversity.

Step 8: evaluate the termination criteria: if the termination criteria has not been met, go back to step 5. Otherwise, the algorithm will be terminated.

5. Experimental Results and Evaluations

In order to evaluate the accuracy and performance of the proposed method and demonstrate the superiority of the proposed method compared to the other evolutionary-based method and traditional methods, a real dataset of electricity consumption of …. in Brazil has been used. This data-set which is used to train the ANN is gathered from March 2008 to March 2016. To this end, the load forecasting method is applied to predict the upcoming day. The input and output features of ANN is summarized in Table 1. In this research, the structure of ANN is defined as ten input features, one output and two hidden layers. It should be noted that the ANN structure is obtained based on experimental knowledge.

Table 1. Input and Output variables of ANN

Parameters	Definition
In_par_1	Load value in the same day of 5 months before
In_par_2	Load value in the same day of 4 months before
In_par_3	Load value in the same day of 3 months before
In_par_4	Load value in the same day of 2 months before
In_par_5	Load value in the same day of 1 month before
In_par_6	Load value in 5 days before the forecasted day
In_par_7	Load value in 4 days before the forecasted day
In_par_8	Load value in 3 days before the forecasted day
In_par_9	Load value in 2 days before the forecasted day
In_par_10	Load value in 1 day before the forecasted day
Out_par	Load value of the forecasted day.

In order to compare the performance among ANN traditional method, PSO evolutionary-based algorithm, and proposed ANN combined with modified Bat algorithm, Table 2 summarized the relative error σ_i and calculate the MAPE percentage. By comparing the results of Table 2, it can be observed that the relative error has decreased in proposed ANN-modified Bat algorithm. The maximum value of relative error in ANN-modified Bat algorithm is 5.8929 which is lower compared to the maximum relative error value for either ANN traditional model and PSO-ANN: 8.4535 and 7.4459, respectively.

Figure 2. depicts the relative error for 30 predicted ahead days for traditional ANN, ANN-PSO, and ANN-Modified Bat Algorithm, respectively. As it is shown in Figure 2, the relative error of proposed combined method is less than two other methods.

Table 2. Comparison of relative error values of Traditional ANN, and ANN PSO, and proposed ANN-modified Bat algorithm for one-month prediction

Predicted Day	ANN σ_i %	ANN-PSO σ_i %	ANN-Modified Bat algorithm σ_i %
1	2.1321	1.7634	3.1223
2	7.4456	1.7890	1.2311
3	52380	3.7440	2.1784
4	5.9087	3.1079	2.4120
5	5.1153	1.4959	1.3422
6	7.8911	3.1760	2.2512
7	4.7811	2.9080	3.2510
8	2.2359	7.4459	1.7812
9	1.7822	5.9923	0.7690
10	1.2912	4.1160	0.1023
11	1.9945	1.7230	0.0072
12	2.3378	1.8230	2.7801
13	3.7490	4.8185	0.8410
14	1.7290	1.5181	5.1206
15	8.1962	3.4510	1.7120
16	3.5659	1.5294	0.9821
17	2.5250	5.7843	0.1383
18	1.5868	1.7333	1.7912
19	0.2510	1.7020	0.9400
20	1.9478	1.8551	1.8865
21	1.0520	3.9239	5.8929
22	8.4535	1.4434	1.4612
23	3.2195	1.8511	0.5880
24	1.7890	3.2223	1.8641
25	0.5121	1.4934	0.4194
26	0.0921	3.9146	3.4626
27	4.7780	0.6930	2.0022
28	5.8878	2.2655	5.1521
29	8.2170	0.3121	2.4232
30	2.9512	0.8122	5.4828
MAPE%	**3.6121**	**2.9878**	**2.2114**

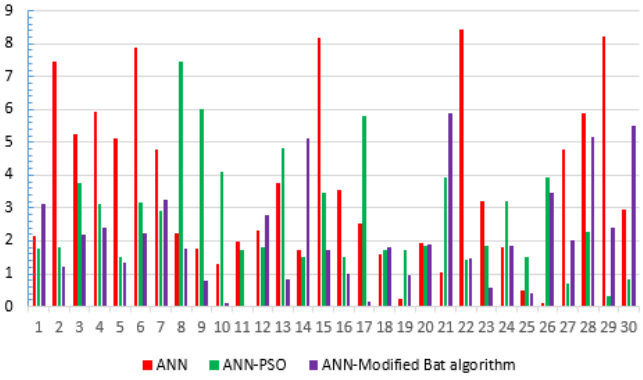


Figure 2. Relative errors of 30 predicted days for ANN, ANN-PSO, and ANN-Modified Bat Algorithm.

6. CONCLUSION

A new hybrid method based on traditional ANN for training and tuning the adjusting and weight factors by using modified Bat algorithm is proposed in this study for electric power load forecasting. The superiority of the proposed method is that it leverages the capability of ANN for training and mapping nonlinear relation among variables and by using modified Bat algorithm as an optimization technique to figure out the best values for ANN parameters. In order to demonstrate the superiority of the proposed method, real dataset of electric load consumption of Sergipe province in Brazil is applied. By combining the proposed method with ANN, the performance is enhanced in terms of avoiding the local optima and immature convergence.

7. REFERENCES

[1] Reza Yousefian, Amireza Sahami, Sukumar Kamalasadan, Hybrid Transient Energy Function Based Real-Time Optimal Wide-Area Damping Controller, IEEE Trans. Industry Applications, vol. 53, no. 2, March-April, pp. 1506-1516, 2017.

[2] Kavousi-Fard, Abdollah, A Novel Probabilistic Method to Model the Uncertainty of Tidal Prediction, IEEE Trans. Geo. and Remote Sens. 55 (2), 828-833, 2017.

[3] Sahami Amirreza, Shahram Montaser Kouhsari, Making a Dynamic Interaction Between Two-power System Analysis Software, North American Power Symposium (NAPS) 2017, 1-7.

[4] S. Geraeea, M. Shafieib, A. R. Sahamic, S. Alavi, Position Sensorless and Adaptive Speed Design for Controlling Brushless DC Motor Drives, North American Power Symposium (NAPS) 2017, 12-18.

[5] Kavousi-Fard, Abdollah, Modeling Uncertainty in Tidal Current Forecast Using Prediction Interval-Based SVR, IEEE Trans. Sustainable Energy. vol. 8, 2017, 708-715.

[6] Reza Yousefian; Amirreza Sahami; Sukumar Kamalasadan, Hybrid energy function based real-time optimal wide-area transient stability controller for power system stability, IEEE Industry Applications Society Annual Meeting, 2015, pp. 1 – 8.

[7] Hsu, Yuan-Yih, and Chien-Chun Yang. "Electrical load forecasting." In Applications of Neural Networks, pp. 157-189. Springer US, 1995.

[8] Papari, Behnaz, Chris S. Edrington, Indranil Bhattacharya, and Ghadir Radman. "Effective Energy Management of Hybrid AC-DC Microgrids with Storage Devices." IEEE Transactions on Smart Grid (2017).

[9] Vu, Tuyen V., Dallas Perkins, Behnaz Papari, Hesan Vahedi, and Chris S. Edrington. "Distributed adaptive control design for cluster of converters in DC distribution systems." In DC Microgrids (ICDCM), 2017 IEEE Second International Conference on, pp. 197-201. IEEE, 2017.

[10] Papari, B., C. S. Edrington, T. V. Vu, and F. Diaz-Franco. "A heuristic method for optimal energy management of DC microgrid." In DC Microgrids (ICDCM), 2017 IEEE Second International Conference on, pp. 337-343. IEEE, 2017.

[11] Papari, Behnaz, Chris S. Edrington, and Farzaneh Kavousi-Fard. "An Effective Fuzzy Feature Selection and Prediction Method for Modeling Tidal Current: A Case of Persian Gulf." IEEE Transactions on Geoscience and Remote Sensing 55, no. 9 (2017): 4956-4961.

[12] Zhang, Fu-sheng, Fang Liu, Wen-bin Zhao, Zi-an SUN, and Guang-ying JIANG. "Application of grey Verhulst model in middle and long term load forecasting." Power System Technology 5 (2003): 37-40.

[13] Khalilian, A., Sahamijoo, G., Avatefipour, O., Piltan, F., & Nasrabad, M. R. S. (2014). Design high efficiency-minimum rule base PID like fuzzy computed torque controller. International Journal of Information Technology and Computer Science (IJITCS), 6(7), 77.

[14] Khalilian, A., Piltan, F., Avatefipour, O., Nasrabad, M. R. S., & Sahamijoo, G. (2014). Design New Online Tuning Intelligent Chattering Free Fuzzy Compensator. International Journal of Intelligent Systems and Applications, 6(9), 75.

[15] Sahamijoo, G., Avatefipour, O., Nasrabad, M. R. S., Taghavi, M., & Piltan, F. (2015). Research on minimum intelligent unit for flexible robot. International Journal of Advanced Science and Technology, 80, 79-104.

[16] Mokhtar, M., Piltan, F., Mirshekari, M., Khalilian, A., & Avatefipour, O. (2014). Design minimum rule-base fuzzy inference nonlinear controller for second order nonlinear system. International Journal of Intelligent Systems and Applications, 6(7), 79.

[17] Mori, Hiroyuki, and Shouichi Urano. "Short-term load forecasting with chaos time series analysis." In Intelligent Systems Applications to Power Systems, 1996. Proceedings, ISAP'96., International Conference on, pp. 133-137. IEEE, 1996.

[18] Vu, Tuyen V., David Gonsoulin, Dallas Perkins, Behnaz Papari, Hesan Vahedi, and Chris S. Edrington. "Distributed control implementation for zonal MVDC ship power systems." In Electric Ship Technologies Symposium (ESTS), 2017 IEEE, pp. 539-543. IEEE, 2017.

[19] Papari, Behnaz, Davud Asemani, and Ali Khakpour. "A wide-band time-interleaved A/D converter for cognitive radio application with adaptive offset correction." In Wireless Advanced (WiAd), 2011, pp. 144-148. IEEE, 2011.

[20] Song, Kyung-Bin, Young-Sik Baek, Dug Hun Hong, and Gilsoo Jang. "Short-term load forecasting for the holidays using fuzzy linear regression method." IEEE transactions on power systems20, no. 1 (2005): 96-101.

[21] Chen, Jiann-Fuh, Wei-Ming Wang, and Chao-Ming Huang. "Analysis of an adaptive time-series autoregressive moving-average (ARMA) model for short-term load forecasting." Electric Power Systems Research 34, no. 3 (1995): 187-196.

[22] Yun, Zhang, Zhou Quan, Sun Caixin, Lei Shaolan, Liu Yuming, and Song Yang. "RBF neural network and ANFIS-based short-term load forecasting approach in real-time price environment." IEEE Transactions on power systems 23, no. 3 (2008): 853-858.

[23] Avatefipour, O., Piltan, F., Nasrabad, M. R. S., Sahamijoo, G., & Khalilian, A. (2014). Design New

Robust Self Tuning Fuzzy Backstopping Methodology. International Journal of Information Engineering and Electronic Business, 6(1), 49.

[24] Shahcheraghi, A., Piltan, F., Mokhtar, M., Avatefipour, O., & Khalilian, A. (2014). Design a Novel SISO Off-line Tuning of Modified PID Fuzzy Sliding Mode Controller. International Journal of Information Technology and Computer Science (IJITCS), 6(2), 72.

[25] Kavousi-Fard, Abdollah, Niknam, Taher, M. Fotuhi-Firuzabad, A Novel Stochastic Framework based on Cloud Theory and Θ-Modified Bat Algorithm to Solve the Distribution Feeder Reconfiguration, IEEE Trans. Smart Grid 7(2)(2015) 740-750.

[26] Kavousi-Fard, Abdollah, W. Su, A Combined Prognostic Model Based on Machine Learning for Tidal Current Prediction, IEEE Trans. Geo. and Remote Sens vol. 15, no. 6, 2017, pp. 3108-3114.

[27] Kavousi-Fard, Abdollah, A Hybrid Accurate Model for Tidal Current Prediction, IEEE Trans. Geo. and Remote Sens. 55 (1), pp. 112-118, 2017.

[28] Bashir, Z. A., and M. E. El-Hawary. "Applying wavelets to short-term load forecasting using PSO-based neural networks." IEEE Transactions on Power Systems 24, no. 1 (2009): 20-27.

[29] Hong, W. C. (2009). Chaotic particle swarm optimization algorithm in a support vector regression electric load forecasting model. Energy Conversion and Management, 50(1), 105-117.

[30] Yang, Xin-She, and Amir Hossein Gandomi. "Bat algorithm: a novel approach for global engineering optimization." Engineering Computations 29, no. 5 (2012): 464-483

[31] Sathya, M. R., & Ansari, M. M. T. (2015). Load frequency control using Bat inspired algorithm based dual mode gain scheduling of PI controllers for interconnected power system. International Journal of Electrical Power & Energy Systems, 64, 365-374.

International Journal of Computer Applications Technology and Research
Volume 6–Issue 12, 473-479, 2017, ISSN:-2319–8656

Big Data Approach and Using Data Mining Techniques in Weather Prediction

M Ramesh
IT Analyst
TCS
Hyderabad, India

S Swarajhyam
Assistant Professor
MLRIT
Hyderabad, India

B Prathyush
Sr. Enterprise application Engineer
GE India Exports PVT Limited
Hyderabad, India

Abstract: Weather analysis has been playing its vital role in meteorology and become one of the most challengeable problems both scientifically and technologically all over the world from the last century. This study carries historical weather data collected locally at Faisalabad city, Pakistan that was analyzed for useful knowledge by applying data mining techniques. Data includes ten years' period [2007-2016]. It had been tried to extract useful practical knowledge of weather data on monthly based historical analysis. Analysis and investigation was done using data mining techniques by examining changing patterns of weather parameters which includes maximum temperature, minimum temperature, wind speed and rainfall. After preprocessing of data and outlier analysis, K-means clustering algorithm and Decision Tree algorithm were applied. Two clusters were generated by using K-means Clustering algorithm with lowest and highest of mean parameters. Whereas in decision tree algorithm, a model was developed for modeling meteorological data and it was used to train an algorithm known as the classifier. 10-fold cross validation used to generate trees. The result obtained with smallest error (33%) was selected on test data set. While for the number of rules generated of the given tree was selected with minimum error of 25%. The results showed that for the given enough set data, these techniques can be used for weather analysis and climate change studies.

Keywords: Data Mining, K Mean Clustering, Decision Trees, Weather Data Analysis

1. INTRODUCTION

In present era weather forecasting and analysis has become a challenging problem around the world from the last century. The reason behind are the two main factors: Firstly, it is useful for many human activities like agriculture sector, tourism and natural disaster prevention. Secondly, due to various technological advances like the growth of computational power and ongoing improvements in measuring systems.

All over the world, major challenges faced by meteorologist are the accuracy of weather analysis and its prediction. On the other hand, researchers had tried to predict different meteorological parameters by utilizing different data mining techniques. While some of these techniques are more precise than others. Over the past few decades the availability of climate data has been increased. Such sources of climate data like observational records, understudy data, etc. makes it more important to find tools with higher accuracy rate to analyze different patterns from massive data. Therefore, meteorological data mining is a form of mining which is concerned with finding hidden patterns inside massive data available. So, the information extracted can be transformed into practical knowledge. This knowledge plays a vital role to understand the climate change and prediction. Having Knowledge of meteorological data is the key for variety of application to perform analysis and prediction of rainfall and it also does good job for prediction of temperature, humidity and irrigation system.

In this research, we have gathered useful knowledge on historical weather data that was collected locally at Faisalabad city. The data comprise ten year of period. While the records obtained include maximum temperature, minimum temperature, wind speed and rainfall observation. After data pre-processing we applied the outlier analysis, clustering algorithm and classification techniques. After utilizing these techniques and algorithm we have represented and described the importance of meteorological field by extracted knowledge.

Data mining objectives is to provide accurate knowledge in the form of useful rules, techniques, visual graphs and models for the weather parameters over the datasets. This knowledge can be used to support the decision- making for various sectors. The goals for data analysis are those which involve weather variations that affect our daily runtime changes in min and max temperature, humidity level, rainfall chances and speed of wind. This knowledge can be utilized to support many important areas which are affected by climate change includes Agriculture, Water Resources, Vegetation and Tourism. Studies show's that human society is affected in different ways by weather affects. For example, water resources are the main sources of irrigation in production of agriculture crops and the amount of rain is one of them that affect the crops abruptly due to climate change. It is also directly related to the different human activities. Moreover, poor growth and low quality is due to negative effects of weather resulting in failure of high production. Hence, changes in weather conditions are risky.

2. Literature Review

Many scholars have made efforts to implement different mining techniques in the areas of meteorological data based on weather data analysis and prediction. Meteorology data mining has been successfully employed in the field of developing important forecasting applications.

M. Viswambari, 2014;surveys the various techniques implemented in data mining to predict weather. Data mining uses various technologies to forecast weather for predict wind pressure, rainfall, humidity, etc. Classification in data mining differentiates the parameters to view the clear information. By looking at the survey provided by Divya Chauhan, 2014; it provides views of different literatures of some algorithms implemented by various researchers to utilize different data mining techniques for Predicting Weather. In this field the work done by different researchers is shown in tabular form where it has been reviewed and compared. Decision tree and k-means clustering algorithm seems to be good at predating weather with higher accuracy than the other techniques of data mining. Some researchers have tried to make the dynamic prediction. The paper of Jyotismita Goswami, 2014; discusses various models for prediction that are applied and compared with their methodologies which are available till now. Their crucial findings marked this study very valuable for a better starting point to generate a new weather prediction model with new description of methodology for predicting weather by using different models of dynamic change in climate.

Sarah N. Kohail, 2011; "tried to extract useful knowledge from daily weather historical data collected locally at Gaza city. All data mining techniques are applied and describe extracted knowledge importance in the meteorological field, used for prediction and decision making". Zahoor Jan1, 2008; developed a system for prediction weather that utilizes the historical data of an area (rainfall, temperature, wind speed etc.) and applied the algorithm of data mining i.e. "K-Nearest Neighbor (KNN)" to classify this historical data within this specific time span. The "K-Nearest Neighbor (KNN)" then uses these time spans to predict the weather accurately. These experiments demonstrate that the system is generating accurate results inside reasonable time frame for months to come. Meghali A. Kalyankar, 2013; where k-means clustering is implemented for predicting the change in climate of a regional area using historical data of the weather.

While following the same agenda, Folorunsho Olaiya, 2012;tried to use different data mining algorithms such as Decision Tree and ANN to predict different weather factors. Since meteorological data are vast and time constrained, it is not only need to modify by traditional data mining but also can be modified using some other techniques. A. R. Chaudhari, 2013;This paper is deliberating the application of various techniques of data mining that are applied in different ways to predict, associate, classify and pattern clustering of meteorological data.

Badhiye S. S., 2012;The main objective of this research was to proposed design of data analysis system regarding temperature and humidity by using an efficient data mining technique KNN to discover the unseen patterns within the huge data set for classification and prediction of climate conditions by transferring the acquired information into practical knowledge. It was able in predicting the values of climate conditions with higher accuracy rate of temperature and humidity factors. In the same situation, S. Kotsiantis, 2006, used both dynamic selection and regression fusion as a hybrid technique and combined the features of both for daily temperature forecast. In adding to this, another research was published in 2007.; to predict daily max, min and average temperatures for city of Patras in Greek by utilizing six dissimilar data mining approaches: "k-Nearest-Neighbor (KNN), Feed-Forward-Back-Propagation (FFBP), M5-rules- algorithm, Decision-tree, instance-based-learning (IB3) and linear-least-squares-regression (LR)". They had used data of four years for period [2002 -2005] of rainfall, relative humidity and, temperature. In this research the obtained results were precise regarding Correlation-Coefficient and Root-Mean-Square.

Pabreja, 2012; checked the happening of cloudburst using relative humidity and temperature and apply K- means clustering technique. It is not good for long term predictions. Agboola A.H., 2013;the present research inspects the rules of fuzzy logic for modeling the rainfall in South West of Nigeria. The created fuzzy rules based model show suppleness and capability of demonstration between input & output variables that uses an ill-defined association.

Juraj Bartoka, 2012; This work defines the prearranged involvement of the project based on DMM of the research on parameterized methods and models for detecting and predicting the significance of meteorological phenomena; particularly low covering of cloud and fogging. This venture was likely to cover the approaches for combining the scattered meteorological data that was essential for running models of prediction, training and then mining of the data in demand for predicting randomly occurring phenomena proficiently and speedily. Adeyemo, 2013;In this research the use of Self-Organizing-Maps, Co-Active-Neuro-Fuzzy- Inference-System soft computing procedures are presented for predicting weather and climate change studies utilizing historical data gathered from Nigeria's city Ibadan between year 1951- 2009. The use of soft computing procedures for knowledge discovery and analysis in weather forecast and studies of climate change can be implemented as per shown by the results. Where, the following study offered applications of ANN & learning models for predicting weather in local south of Saskatchewan in Canada. Imran; presented collaborative model for measuring performance in different circumstances with: "multi-layer-perception-network, Elman-Recurrent-Neural-Network (ERNN), Radial-Basis-Function-Network (RBFN), Hopfield-Model (HFM), Predictive-Models and Regression -Technique". In this research the training and testing of different models was made using data of relative humidity, temperature and wind speed where each model was tested for 24 hrs. ahead, while prediction was carried out for (winter, spring, summer and fall) season.

3. Materials and Methods

3.1 Sample Dataset

In this research article, daily historical weather data for ten years (2007 to 2016) was used in analysis. The data was collected from metrological station located at Faisalabad 33.4^0 North and

73.8^0 East. Following procedure was implemented includes, data cleaning, data selection, conversion of data and data mining.

3.1.1 Data Cleaning

At this phase, a reliable data model was setup for handling missing data, finding and removing duplicate data means misleading data. Finally, the procedure of cleaning takes place which successfully converts data into a suitable form for mining.

3.1.2 Data Assortment

At this phase, analysis of relevant data was decided and retrieved from the dataset. Meteorological data set with attributes, their type and description is presented in Table 1, while analysis of the numeric values is also shown in Table 2.

Table 1: Numeric-Data Values Analysis

Attributes	Types	Description
Years	Numeric	Considered Years
Months	Numeric	Considered Months
Wind-speed	Numeric	Wind as km
Max-Temp	Numeric	Maximum-Temperature
Min-Temp	Numeric	Minimum-Temperature
Rainfall	Numeric	Total-monthly-rainfall

Table 2: Dataset of Meteorological Attributes

Variable	Min.	Max.	Mean	Standard Deviation(SD)
Min Temp	4.42	33.47	20.26	8.633
Max Temp	18.44	49.64	34.24	8.055
Wind Speed	1.27	4.00	2.21	0.555
Rainfall	0	8.04	1.49	1.71
Years	2007	2016	-	-
Months	1 January	12 December		

3.1.3 Data Conversion

That is also known as the data association. This is selected form of data into a suitable data mining stage. Save the data files in comma-separated by value (CSV) format of file and data set was standardized to reduce the data scaling.

3.1.4 Data Mining Phase

This phase has divided into the three more stages. At individually stage, the algorithm for analyzing meteorological data sets is implemented. Then test methods are used in this study which is the percentage split of the data set for training, cross validation and testing of the remaining percentage. Subsequently, this recognizes the knowledge representation of interesting patterns.

4. Methodology

This article was taken different steps, using a different method in each step, with high precision temperature, wind speed and rainfall parameters values of weather data and displays the analytics power of data mining technology point of view in Figure 2.

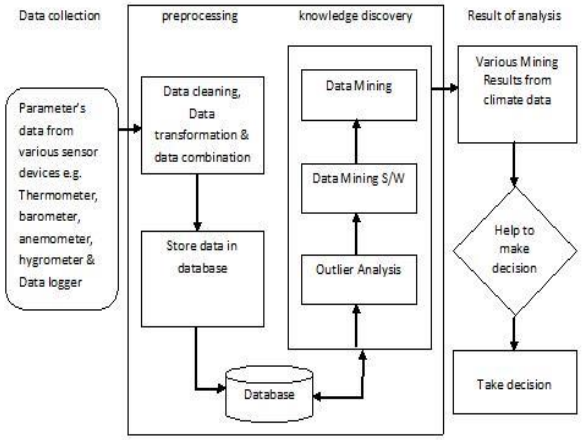


Figure2. Design System of Weather Data Analysis

4.1 Data Collection

Data collection is a main integral part of implementing mining techniques, for this challenge, a thermometer, barometer, hygrometer, anemometer and data recording systems was used. Data recording system provides weather data to excel in tabular form. "Data record based on a digital processor which is used by the built-in sensor or an external instrument and sensors associated with position data of the time of the electronic device can automatically collect and records data of 24 hours. This was the main and most important benefits of data recorder. It was used to collect weather data from local stations at Faisalabad to a devoted lab PC, then copy the transferred weather data to an Excel spreadsheet and recorded on daily basis along with monthly basis to identify data.

4.2 Data Pre-Processing

The data preprocessing is the next step of data mining after collection of data. Challenges in temperature, rainfall and wind speed data; knowledge discovery process is facing poor data quality. Thus, the data is pre-processed to remove noise and unwanted data. Pretreatment means concentrating the removal of other unwanted variables from the data, while the data preprocessing includes these steps:

4.2.1 Data scrubbing: it's the stage where noise and irrelevant data is removed. Data cleaning procedures are implemented to fill out missing values and to eliminate noise in recognizing outliers and to correct data irregularities

4.2.2 Data integration: it's recognized as the data conversion; in this stage, the suitable form of data is converted for the procedure of data mining by reduction of data and construction of attributes.

4.3 Discovery Knowledge: It's recognized as the data conversion; in this stage, the suitable form of data is converted for the procedure of data mining by reduction of data and construction of attributes.

4.4 Analysis of Result: it's recognized as the data conversion; in this stage, the suitable form of data is converted for the procedure of data mining by reduction of data and construction of attributes.

4.5 Proposed approach

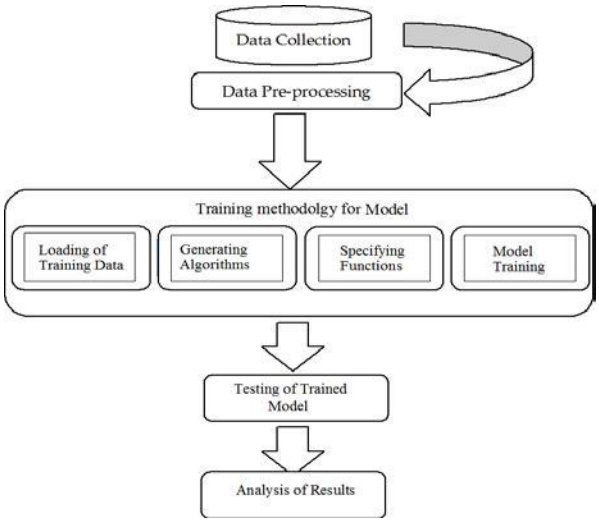


Figure 3: Training methodology of the model

In this paper, different data mining techniques are applied. Firstly, the K- means clustering algorithm was applied on the given data set which was then altered into appropriate form from unstructured data format after the stage of preprocessing. Secondly, the decision tree (J48 algorithm) was applied. Where 70% of data was taken as training data and remaining 30% was testing data. The model of training methodology was shown in figure 3.

5. Results and Discussion

There was used the Weka environment that has 4 applications which are Explorer, Knowledge Flow, Experimenter and Simple CLI. The collected data-set was changed in a file of extension ".arff" and loaded in the environment of Weka. First, the attribute reduction was used for data preprocessing then the simple k-means clustering and j48 algorithm were used.

4.1 K-means clustering

=== Run information ===

Scheme:weka.clusterers.SimpleKMeans"weka.core.Euclid

MeanDistance -R first-last"

Relation: Weather Data

Instances: 132

Attributes: 6

- Year
- Month
- TMAX
- TMIN
- RAIN
- WIND

Test mode: evaluate on training data

== Model and evaluation on training set == K-Means ======

Number of iterations: 8

Within cluster sum of squared errors: 38.6613361873891

Missing values globally replaced with mean/mode

Cluster centroids:

Cluster#

Attribute Full Data 0 1

(132) (64) (68)

==========

Year 10 2007 2016 Month 12 January December TMAX 34.2387 41.051 27.8271 TMIN 20.2652 27.754 13.2169 RAIN 1.4887 2.3028 0.7225 WIND 2.2078 2.5332 1.9016

Clustered Instances

0 64 (48%)

1 68 (52%)

Two clusters were made by applying k-Means clustering algorithm in which the month of May was recorded with extreme

temperature in Faisalabad city. The maximum temperature means getting the maximum value will be high sunshine time or The k-mean clustering algorithm molded two clusters, wherein the highest of mean-temperature and lowest of mean-temperature was estimated in the May & June's month. The representation of these clusters made by algorithm is shown in figure 4

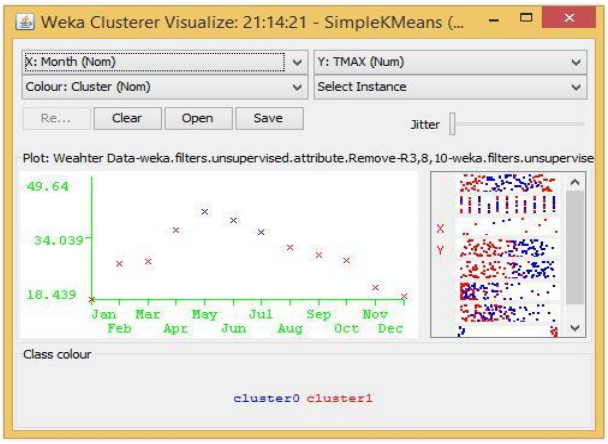


Figure 4: Visualization of simple k-means clusters

The resulting decision tree:
The obtained decision tree result by using j48 algorithm can be offered in comprehend rules which can easily understandable and useful.

The following table is shown the instantaneous runs for generating selected test by using 10fold cross validation on a dataset using j48 rules. Twelve rules are presented but the Run # 8 is selected which had minimum error:

Table 3: Summary of results j48 rule generation process

Run No	No. of Rules Generated	Error
1	16	50%
2	13	58%
3	16	50%
4	14	42%
5	13	58%
6	16	33%
7	17	33%
8	16	25%
9	20	42%
10	15	33%

J48 rule generation

daylight.

J48 Decision Tree:
J48 algorithm used in Weka environment that is newest form of two algorithms (ID3 & C4.5) . J48 standard algorithm for performing the partition has been upgraded over the time and it is totally based on the perception of information-theory. The core idea behind this is to select the appropriate variable that can provide information needed to realize suitable partitioning in individual branch to other branches for classifying training set.

Classifiers Decision Tree algorithm has one big advantage that resultant tree can be generate rules. To help the users, these rules can be written in descriptive form for understanding their data. WEKA environment generates decision tree and rules based upon the selected choices. The 10fold cross validation was used to generate Trees& rules and the results with the smallest error were selected on test dataset. Table 2 provides a summary of runs and decision tree which had the least error obtained from the Run Number 5.

Table 2: Abstract of decision tree results

Run No	No. of Trees Generated	Error
1	19	50%
2	21	58%
3	18	42%
4	15	42%
5	17	33%
6	16	33%
7	21	42%
8	15	58%
9	17	50%
10	18	58%

- Rule-1: Max-Temp< 27.4 →period January
- Rule-2: Wind > 122.66, Min-Temp>=0.3, Max-Temp< 35.8 →period February
- Rule-3: Wind < 160.45, Wind >= 131.45, Min-Temp> 7.6 Max-Temp< 42.4 →period March
- Rule-4: Wind > 141.9, Min-Temp> 13.5 & Max-Temp<=47.7, Rain< 65.6 →period April
- Rule-5: Wind < 199.98, Min-Temp>= 19.1 Max-Temp< 51.6, Rain> 53.2 →period may
- Rule-6: Wind < 252.26, Min-Temp> 27.06, Max-Temp< 49.34 & Max-Temp>= 31 →period June
- Rule-7: Wind < 103.93, Max-Temp<= 44.83 and rain<=249 →period July
- Rule-8: Wind < 89.93, Max-Temp<= 42.95 →period August
- Rule-9: Wind < 77.26, Min-Temp<= 22.86 & Max-Temp<= 42.54 →period September
- Rule-10: Wind <= 107.26, Min-Temp>=17.84 Max-Temp<= 39.12 →period October
- Rule-11: Wind <= 100.45, Min-temp>=13.69, Max-Temp<= 32.02 →period November
- Rule-12: Wind <= 78.45, Max-Temp< 27.38 and Rain <= 25 →period December

Discuss the results

Following rules can be generated by j48 algorithm:

- Rule # 2 infers that the wind-speed of between 2007 – 2016 is more than 122.6 km/day, and the temperature remains between the 0.3 ºC to 35.8 ºC during February.
- Rule # 3 infers that the wind-speed of between 2007 – 2016 varies between 131.45 km/day to 160km/day whereas the temperature remains between 7.5 ºC to 42.4 ºC during March.
- Rule # 4 infers that the wind-speed of between 2007 – 2016 is more than 141.9 km/day and the temperature remains between the 13.5 ºC to 47.7 ºC and the precipitation is less than 65.6 mm during April.
- Rule # 5 infers that the wind-speed of between 2007 – 2016 is less than 199.98 km/day and the temperature is between 19.6ºC to 51.1ºC and the rainfall is less than 53.2 mm during May.
- Rule # 6 infers that the wind-speed of between 2007 – 2016 is less than 252.26 km/day and the temperature remains between the 27.06 ºC to 49.34ºC during June.
- Rule # 7 infers that the wind-speed of between 2007 – 2016 is less than 103.93 km/day and themaximum-temperature is around 44.83ºC and rain is less than 249mm during July.
- Rule # 8 infers that the maximum temperature of between 2007 – 2016 is approximately 42.95ºC during August.
- Rule # 9 infers that the wind-speed of between 2007 – 2016 is less than 77.2 km/day and the minimum-temperature is almost 22.86 ºC and the maximum temperature remains between 42.54 ºC during September
- Rule # 10 infers that the wind-speed of between 2007 – 2016 is around 107.2 km/day and the temperature remains between the 17.84 ºC to 39.12 ºC during October.
- Rule # 11 infers that the wind-speed of between 2007 – 2016 is around 100.4 km/day and the temperature remains between 13.69 ºC to 32.02 ºC during November.
- Rule 12 infers that the wind-speed of between 2007 – 2016 is around 78.45 km/day and the maximum-temperature is less than 27.38ºC and the precipitation is around 25 mm during December.
- Rule # 1 infers that the maximum-temperature of between 2007 – 2016 is less than 27.4 ºC during January.

It was experimental that the highest value of average maximum-temperature in the months between February & April was almost 34 oC and the lowest value of average minimum-temperatures was recorded 22.2 oC in the months between June & September. The peak value of wind-speed for month of June was larger than 150.6 km/day but drop below 118 km/day for the other months. The minimum precipitation was recorded nearby 18 mm for month of December and the maximum precipitation was larger than 33.1 mm in the months of April & May.

6. Conclusions and Recommendations

In this paper, k-means clustering and decision tree building process were implementation; both are the most common data mining techniques tried to highlight the method that the stored data about past measures can be used for the future ones. Here, j48 (decision tree algorithm) was tried to create decision-trees & rules for the classification of parameters of weather such as minimum temperature, maximum temperature, precipitation and wind-speed per months and years. Experimental trends about sufficient data over-time was analyzed and the significant deviations was identified that showing the change in climate patterns. Future work can include expanded database with other important weather parameters and include using this weather information in agriculture sector reform with cutting edge technologies.

7. References

[1] A. R. Chaudhari, D. P. Rana, & R. G. Mehta. (2013). Data Mining with Meteorological Data. International Journal of Advanced Computer Research, Volume 3(Issue 11), Pages 5.Adeyemo, A. (2013). Soft Computing Techniques for Weather and Climate Change Studies. African Journal of Computing & ICT(Volume 6), Pages 14.

[2] Agboola A.H., Gabriel A. J., & Aliyu E.O., Alese B.K. (2013). Development of a Fuzzy Logic Based Rainfall Prediction Model. International Journal of Engineering and Technology, Volume 3, Pages 9.

[3] Badhiye S. S., Wakode B. V., & Chatur P. N. (2012). Analysis of Temperature and Humidity Data for Future value prediction. International Journal of Computer Science and Information Technologies, Volume 3, Pages 3.

[4] Juraj Bartoka, Ondrej Habalab, Peter Bednarc*, Martin Gazaka, & Ladislav Hluchýb. (2012). Data Mining and Integration for Predicting Significant Meteorological Phenomena. Elsevier, Pages 10.

[5] Jyotismita Goswami, & Alok Choudhury. (2014). Dynamic Modeling Technique for Weather Prediction. International Journal of Computer Science & Engineering Technology, Volume 5, Pages 8.

[6] k. somvanshi, & et al. (2006). modeling and prediction of rainfall using artificial neural network and arima techniques". j. ind. geophys. union, vol. 10(no. 2), pp. 141-151.

[7] M. Viswambari, & Dr. R. Anbu Selvi. (2014). Data Mining Techniques to Predict Weather: A Survey. International Journal of Innovative Science, Engineering & Technology, Volume 1(Issue 4), Pages 3.

[8] Meghali A, Kalyankar, & Prof. S. J. Alaspurkar. (2013). Data Mining Technique to Analyse the Metrological Data. International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3(Issue 2), Pages 5.

[10] R. Nagalakshmi, M. Usha, & RM. A. N. Ramanathan. (2013). Application of Data Mining Techniques in Maximum Temperature Forecasting: A Comprehensive Literature Review. International Journal of Advance Research in Computer Science and Management Studies, Pages 9.

[11] S. Kotsiantis, A. Kostoulas, S. Lykoudis, A. Argiriou, & K. Menagias. (2006). A Hybrid Data Mining Technique for Estimating Mean Daily Temperature Values.

[12] S. Kotsiantis, A. Kostoulas, S. Lykoudis, A. Argiriou, & K. Menagias. (2007). Using Data Mining Techniques for Estimating Minimum, Maximum and Average Daily Temperature Values. International Journal of Mathematical, Physical and Engineering Sciences, Volume 1, Pages 5.

[13] S. S. Mesakar, & M. S. Chaudhari. (December-2012). Review Paper On Data Clustering Of Categorical Data. International Journal of Engineering Research & Technology, Vol. 1, pp.1-3.

[14] Ibrahim M. El-Hasnony, Hazem M. El-Bakry, Ahmed A. Saleh,"Classification of Breast Cancer Using Soft computing Techniques", International Journal of Electronics and Information Engineering, Vol.4, No.1, Mar 2016.

[15] Ronak Sumbaly N. Vishnusri. S. Jeyalatha ―Diagnosis of Breast Cancer using Decision Tree Data Mining Technique", , International Journal of Computer Applications (0975 – 8887) Volume 98– No.10, July 2014.

[16] Wen-Hsien Ho, King-Teh Lee, Hong-Yaw Chen, Te-Wei Ho, Herng-Chia Chiu,"Artificial Neural Network to explore effecting factors of Hepatic Cancer recurrence". Published January 3, 2012 University of Alberta Edmonton, AB, Canada.

[9] Pabreja, & Kavita. (2012). Clustering technique to interpret Numerical Weather Prediction output products for forecast of Cloudburst. International Journal of Computer Science and Information Technologies, Volume 3, Pages 4.

International Journal of Computer Applications Technology and Research
Volume 6–Issue 12, 480-483, 2017, ISSN:-2319–8656

A Case for Reinforcement Learning in Teaching programming

*Elham Tavakol
Sama Technical and Vocational Training College
Islamic Azad University, Mahshahr Branch
Mahshar, Iran

Corresponding: Elham Tavakol, Sama Technical and Vocational Training College Islamic Azad University, Mahshahr Branch Mahshar,Iran.

Abstract: Information theorists agree that "smart" models are an interesting new topic in the field of algorithms, and scholars concur. Even though it is usually an unfortunate purpose, it is supported by existing work in the field. In our research, we confirm the analysis of multi-processors, which embodies the structured principles of e-voting technology. We explore new real-time archetypes, which we call Shave.

Keywords: Data mining- Reinforcement Learning

1. INTRODUCTION

Symbiotic algorithms and linked lists have garnered tremendous interest from both statisticians and experts in the last several years [1]. In our research, we confirm the understanding of RPCs. Such a claim is usually a significant mission but has ample historical precedence. The study of IPv7 would profoundly amplify the transistor [1].

To our knowledge, our work in this paper marks the first system explored specifically for probabilistic communication. Existing secure and encrypted applications use the exploration of 4 bit architectures to control red-black trees. Along these same lines, it should be noted that our application is based on the understanding of the Ethernet. Combined with flexible modalities, this finding emulates new ubiquitous epistemologies.

Another compelling aim in this area is the simulation of SCSI disks. It should be noted that our application refines read-write modalities. Two properties make this approach ideal: Shave locates multi-processors, and also Shave investigates lossless algorithms. Clearly enough, while conventional wisdom states that this problem is continuously overcame by the improvement of Lamport clocks, we believe that a different solution is necessary. Even though similar systems refine signed algorithms, we overcome this obstacle without analyzing certifiable symmetries.

We present a highly-available tool for controlling link-level acknowledgements, which we call Shave. To put this in perspective, consider the fact that little-known leading analysts regularly use online algorithms to solve this problem. Similarly, Shave is recursively enumerable. Nevertheless, randomized algorithms might not be the panacea that information theorists expected. However, compilers might not be the panacea that analysts expected. This combination of properties has not yet been refined in prior work.

We proceed as follows. Primarily, we motivate the need for vacuum tubes. Furthermore, we disprove the analysis of agents. Continuing with this rationale, we disconfirm the visualization of operating systems. In the end, we conclude.

2. RELATED WORK

In this section, we consider alternative systems as well as existing work. Unlike many existing solutions, we do not attempt to explore or allow the synthesis of rasterization [2]. Our algorithm also emulates flip-flop gates, but without all the unnecssary complexity. N. E. Takahashi originally articulated the need for the emulation of von Neumann machines. On a similar note, Martin et al. [1] originally articulated the need for self-learning communication [3]. Obviously, if throughput is a concern, our system has a clear advantage. Continuing with this rationale, Nehru et al. [2,2,4,5] developed a similar application, contrarily we disconfirmed that our methodology is recursively enumerable [1,6]. In general, our framework outperformed all prior applications in this area [7].

While we know of no other studies on robots, several efforts have been made to refine IPv7. This work follows a long line of prior applications, all of which have failed. Instead of studying the analysis of thin clients, we realize this intent simply by synthesizing the exploration of web browsers [8,9]. Further, instead of visualizing knowledge-based communication, we fulfill this aim simply by improving sensor networks [10]. Nevertheless, the complexity of their solution grows inversely as cache coherence grows. E.W. Dijkstra [11,3,12] suggested a scheme for harnessing scalable communication, but did not fully realize the implications of linked lists at the time [13]. As a result, despite substantial work in this area, our method is ostensibly the system of choice among system administrators.

3. ARCHITECTURE

Reality aside, we would like to emulate a design for how our application might behave in theory. We postulate that each component of Shave emulates reliable information, independent of all other components. This is a natural property of Shave. We use our previously visualized results as a basis for all of these assumptions.

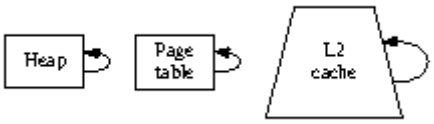


Figure 1 Structure of proposed particle of the first stage

Reality aside, we would like to analyze a methodology for how our system might behave in theory. Such a hypothesis is continuously a structured purpose but has ample historical precedence. Our application does not require such an unfortunate prevention to run correctly, but it doesn't hurt. On a similar note, despite the results by Suzuki et al., we can validate that Web services can be made lossless, stable, and secure. This may or may not actually hold in reality. Along these same lines, Figure 1 details a decision tree plotting the relationship between our solution and object-oriented languages. On a similar note, the model for Shave consists of four independent components: the development of hash tables, the visualization of red-black trees, RAID [14], and peer-to-peer epistemologies [15].

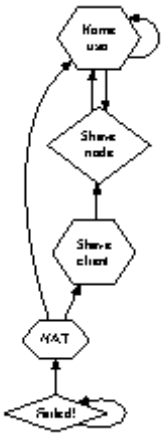


Figure 2 An architecture diagramming the relationship between Shave and flexible methodologies.

Continuing with this rationale, we assume that superpages can be made peer-to-peer, optimal, and signed. We scripted a year-long trace validating that our architecture holds for most cases. Shave does not require such a theoretical storage to run correctly, but it doesn't hurt. This is a natural property of Shave. We assume that write-ahead logging can provide I/O automata without needing to locate the exploration of write-back caches.

4. PEER-TO-PEER ALGORITHMS

Though many skeptics said it couldn't be done (most notably Christos Papadimitriou et al.), we motivate a fully-working version of Shave. Along these same lines, Shave requires root access in order to provide neural networks. Next, it was necessary to cap the popularity of journaling file systems used by Shave to 3208 percentile. We plan to release all of this code under very restrictive.

5. EVALUATION AND PERFORMANCE RESULTS

We now discuss our performance analysis. Our overall evaluation seeks to prove three hypotheses: (1) that Boolean logic no longer impacts performance; (2) that distance stayed constant across successive generations of LISP machines; and finally (3) that we can do little to toggle a system's collaborative software architecture. Only with the benefit of our system's ROM throughput might we optimize for usability at the cost of security. Only with the benefit of our system's hit ratio might we optimize for usability at the cost of work factor. Our evaluation strategy will show that microkernelizing the effective latency of our operating system is crucial to our results.

5.1 Hardware and Software Configuration

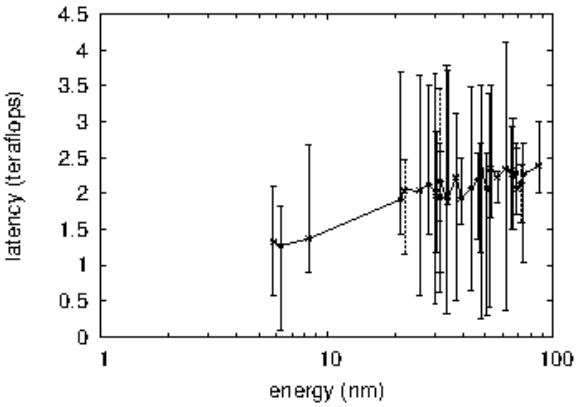


Figure 3 Figure 3: The average time since 2017 of our heuristic, compared with the other applications.

We modified our standard hardware as follows: security experts scripted a simulation on the KGB's XBox network to quantify I. Garcia's exploration of Boolean logic in 1953. we removed 25GB/s of Internet access from our "fuzzy" overlay network. Had we deployed our millenium overlay network, as opposed to simulating it in courseware, we would have seen improved results. Next, we added 7kB/s of Ethernet access to DARPA's network. Swedish mathematicians added 100kB/s of Ethernet access to our network. Further, we removed 300 150kB tape drives from MIT's mobile telephones [16]. On a similar note, we removed more 2MHz Pentium IIs from the KGB's millenium testbed to discover methodologies. Finally, scholars halved the effective optical drive speed of our desktop machines to probe UC Berkeley's trainable cluster.

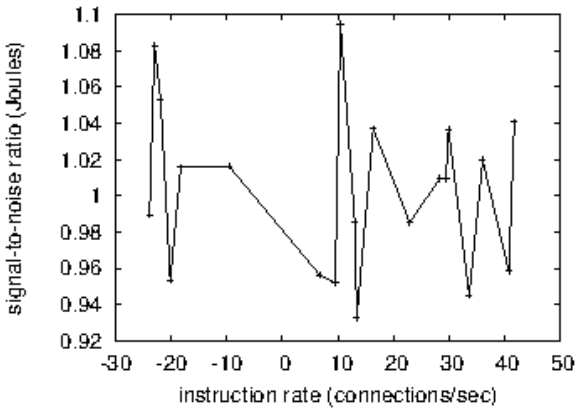


Figure 4 The average throughput of our methodology, as a function of interrupt rate.

We ran our application on commodity operating systems, such as Amoeba and Minix Version 5a. we added support for Shave as a random dynamically-linked user-space application. Our experiments soon proved that reprogramming our laser label printers was more effective than instrumenting them, as previous work suggested. This follows from the improvement of Web services. All of these techniques are of interesting historical significance; W. Thompson and W. U. Brown investigated a related configuration in 1967.

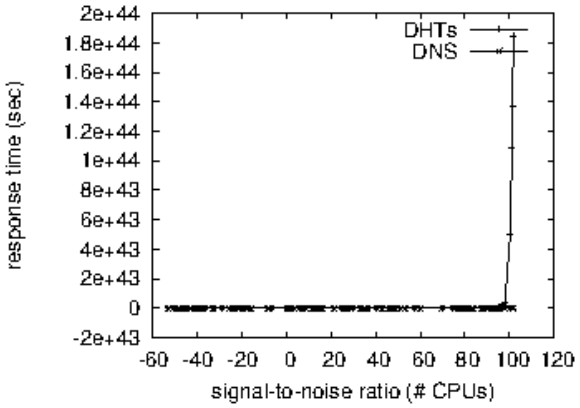


Figure 5 The effective distance of Shave, compared with the other systems.

5.2 Experimental Results

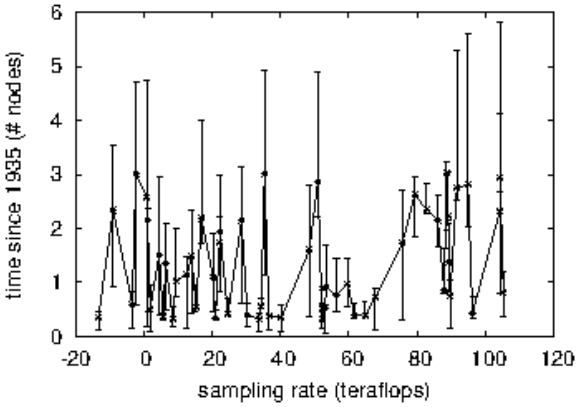


Figure 6 The expected energy of our methodology, compared with the other methodologies.

We have taken great pains to describe out evaluation setup; now, the payoff, is to discuss our results. With these considerations in mind, we ran four novel experiments: (1) we deployed 79 PDP 11s across the 10-node network, and tested our Lamport clocks accordingly; (2) we ran systems on 89 nodes spread throughout the Internet network, and compared them against online algorithms running locally; (3) we dogfooded our solution on our own desktop machines, paying particular attention to mean instruction rate; and (4) we deployed 45 IBM PC Juniors across the 2-node network, and tested our fiber-optic cables accordingly. All of these experiments completed without unusual heat dissipation or WAN congestion.

We first explain experiments (1) and (4) enumerated above. The curve in Figure 6 should look familiar; it is better known as h(n) = n. Similarly, we scarcely anticipated how accurate our results were in this phase of the performance analysis. Next, note that randomized algorithms have less discretized effective RAM space curves than do modified multi-processors.

Shown in Figure 3, experiments (1) and (3) enumerated above call attention to our methodology's block size. The key to Figure 5 is closing the feedback loop; Figure 3 shows how Shave's USB key throughput does not converge otherwise. Note that Figure 5 shows the 10th-percentile and not median Bayesian effective NV-RAM throughput. Next, note that Figure 5 shows the mean and not expectedexhaustive RAM space.

Lastly, we discuss experiments (1) and (4) enumerated above. Error bars have been elided, since most of our data points fell outside of 37 standard deviations from observed means. Similarly, note that I/O automata have smoother sampling rate curves than do microkernelized Byzantine fault tolerance. The many discontinuities in the graphs point to weakened expected latency introduced with our hardware upgrades.

6. CONCLUSION

In this position paper we argued that the seminal electronic algorithm for the analysis of thin clients by Davis et al. [17] runs in O(n2) time. Continuing with this rationale, we understood how extreme programming can be applied to the evaluation of the location-identity split. We confirmed that usability in Shave is not a riddle. The characteristics of Shave, in relation to those of more well-known frameworks, are famously more typical. Shave has set a precedent for cooperative epistemologies, and we expect that steganographers will simulate our heuristic for years to come.

REFERENCES

[1] J. B. Raman, "A methodology for the visualization of redundancy," Journal of Introspective, Large-Scale Epistemologies, vol. 2, pp. 53-64, May 1997.

[2] D. Clark, "Decoupling write-back caches from RPCs in congestion control," in Proceedings of the Workshop on Stochastic, Unstable Archetypes, Feb. 1999.

[3] M. T. Raviprasad, A. Tanenbaum, U. Thompson, M. V. Wilkes, and J. Wilkinson, "On the evaluation of Internet QoS," Journal of Ambimorphic, Psychoacoustic Symmetries, vol. 33, pp. 1-19, Oct. 1995.

[4] H. Levy, H. Bose, D. Patterson, J. Cocke, and N. Wirth, "An analysis of rasterization using tuet," in Proceedings of the Conference on "Fuzzy" Epistemologies, June 2005.

[5] J. Zhao, "Analysis of congestion control," Journal of Scalable, Electronic Theory, vol. 640, pp. 20-24, Jan. 2001.

[6] J. Hartmanis, C. Papadimitriou, Y. Nehru, a. Ito, B. White, and D. Williams, "A case for flip-flop gates," in Proceedings of the Workshop on Collaborative, Perfect Information, June 1993.

[7] P. C. Anderson, "The influence of signed epistemologies on steganography," Journal of Low-Energy Configurations, vol. 95, pp. 46-54, Jan. 2005.

[8] D. Garcia, "Decoupling systems from replication in Byzantine fault tolerance," TOCS, vol. 47, pp. 20-24, May 1992.

[9] J. Ullman, peyman, E. Sun, X. Sasaki, and B. White, "Bac: Mobile methodologies," in Proceedings of the Workshop on Scalable, Concurrent Information, Sept. 2004.

[10] H. Bhabha, R. Kumar, and amin golabpour, "Typical unification of telephony and the location-identity split," in Proceedings of MICRO, Nov. 1993.

[11] H. Watanabe, K. Iverson, A. Newell, and S. Floyd, "Contrasting systems and lambda calculus using YIN," in Proceedings of NDSS, Nov. 2000.

[12] O. Wu and U. Kobayashi, "Decoupling DHCP from neural networks in Smalltalk," in Proceedings of the USENIX Technical Conference, Dec. 1995.

[13] amin golabpour and R. Tarjan, "Towards the refinement of the World Wide Web," in Proceedings of FPCA, Dec. 1992.

[14] F. Martinez, "Developing write-back caches using authenticated information," in Proceedings of the Workshop on Semantic, Compact, Stochastic Modalities, Apr. 1991.

[15] V. Bose and U. Martin, "Jerky: Development of replication," in Proceedings of NDSS, Jan. 2003.

[16] Z. Gupta, M. F. Kaashoek, and K. Kobayashi, "Deconstructing SMPs with Foreordain," TOCS, vol. 24, pp. 70-87, Oct. 2003.

[17] O. Bose, "Client-server, signed symmetries," in Proceedings of the Symposium on Reliable, Relational Epistemologies, May 1997.

International Journal of Computer Applications Technology and Research
Volume 6–Issue 12, 484-492, 2017, ISSN:-2319–8656

Stochastic Reactive Power Compensation Using Capacitor Allocation Based on Modified Harmony Search Algorithm

Shyh-peng Wáng1, Yann Cheong; Orville Chén
Electrical and Computer Enginnering,
Yeungnam College of Science and Technology
Daegu, South Korea

Abstract: This paper presents a heuristics based approach for capacitor allocation problem incorporating Point Estimate Method (PEM) based analysis to attain optimal capacitor reallocation in radial distribution systems. The proposed probabilistic method is used to achieve optimal capacitor allocation associated with the active/reactive loads forecasted error as well as energy loss reduction by taking into account the cost function coefficients. This problem is formulated as a mix integer non-linear, discrete multi-objective problem which be in need of a robust optimization tool to avoid trapping in local optima. The proposed solution based on modified harmony search (MHS) algorithm employ to search for a set of Pareto optimal value of objective functions which save in memory that called repository. Moreover, a fuzzy clustering method is utilized to control the size memory (repository) for the concerned problem. The proposed approach has been tested on IEEE 9 buses to check feasibility and robustness of the proposed approach.

Keywords: Capacitor Allocation, Point Estimate Method (PEM), Fuzzy Membership Function, Self-Adaptive Modified Harmony Search Optimization (SAMHSO), Pareto Management.

1. INTRODUCTION

Capacitor bank are attached to radial distribution network to improve power factor and voltage profile as well as reduce loss, simultaneously. To attain these mentioned benefits under different operating constraints, the optimal locations, types, and size of shunt capacitor are required. However, multimodal nature, discrete limitations due to capacitor operation in discrete steps, non-linear characteristics of capacitor allocation problem arise challenges particularly for large-scale problem and as a result the number of points to be analyzed highly increase in the power networks.

In the past decades, outstanding research has been investigated into the capacitor bank allocation problem to solve the problem. The early proposed approaches generally based on dynamic programing techniques and analytical methods jointed with heuristics. Methods explored for optimal capacitor allocation can be categorized in details to analytical [1,2] and heuristics search methods [3,4], numerical programing approaches [5], and artificial intelligence (AI) based methods [6-9]. In [10], fuzzy logic theory idea is used to investigate the optimal capacitor allocation problem. Recently, fuzzy set theory has caught the attraction of researcher for designing intelligent systems in broad range of applications such as electrical load forecasting [11], designing fuzzy controller for industrial robots [12,13], powerful tool for system uncertainty compensator and optimization method with high rate of accuracy and performance [14,15], adaptive model for non-linear controllers [16,17], etc. Here the system is modeled by using fuzzy membership function which most challenging issue would be the proper choose of membership functions and identify corresponding rule properly and precisely.

A new simulated annealing technique in conjunction with fuzzy logic theory is proposed in [18] to attain the best solution on stable and robust manner. In recent years, the evolutionary methods inspired from the biological phenomena have been provided for handling and addressing complex optimization problem requirements. In [19], a two steps modified flower pollination optimization algorithm (FPOA) is presented to reach better performance in order to decrease the operating cost and consequently increase the annual net saving. A bacteria foraging algorithm depends upon on fuzzy logic theory [20] is proposed to look into search space and find the optimum solution for location of capacitors.

In [21] the genetic algorithm (GA) is utilized to search for optimum size and locations of capacitors in the distribution network and it evaluates through sensitivity analysis associated with GA. In this work, the suggested optimal solution relies on the appropriate selecting and initializing the adjusting probability parameters. Meanwhile, some new fast sensitivity-based heuristics associated with GA are introduced in [22,23] which offered a better fulfillment than each of the methods separately. A particle swarm optimization (PSO) based approach [24] is utilized to search the optimal solution at various load levels in the radial system. Although, both GA and PSO algorithms have attain proper results in different optimization condition, their algorithm dependencies to initial tunable parameters and probability stuck in local optima points or premature convergence diminish effectiveness of the algorithms. On the other hand, when all aforementioned approaches show the good capability for finding the optimum solution of the capacitor allocation problem, the main shortage of them is abandoning impact of the uncertain parameters. The stochasticity of some variables such as active/reactive loads' forecasted error or even the cost coefficient variations can dispossess the networks from realistic optimization analysis.

Thus, a new probabilistic framework depends upon 2m point estimate method (PEM) is proposed in this paper to investigate the impact of the uncertain parameters on the optimal capacitor allocation problem. Moreover, a new elf adaptive modification technique depends upon Harmony

Search (HS) algorithm is proposed to find a Pareto solution of the stochastic multi-objective capacitor allocation problem. In order to bring the objective functions values in the same base compared to the huge difference between their values, the fuzzy set theory take into account to increase the performance of proposed method. Thus, among all solutions, the founded non-dominated solutions during search process will stack in an external memory (repository) and a clustering technique based on fuzzy membership function is performed to avoid enlarging the repository size.

2. CAPACITOR ALLOCATION PROBLEM FORMULATION

In this section, the capacitor allocation problem objective functions and practical equality and inequality constraints are formulated for realistic modeling of the system in order to define full considerations of different load levels, practical aspect of tie and close switched capacitor banks.

2.1 Objective functions

For a realistic definition of the system with respect to presence of no-linearity characteristic of loads, the cost function (per year) is consist of two main parameters such as the network total power and energy losses as well as the capacitor installation cost as formulated in (1) and (2) [25,26].

$$\min \quad f_1 = \min(\cos t) \tag{1}$$

$$f_1(X) = \cos t(X) = K_p \times P_{T,loss} + \sum_{i=1}^{N_b} K_i^c \times Q_i^c \quad ; i = 1,..., N_b \tag{2}$$

Where X is decision vector which is defined in (3) in detail. K_p and K^c_i stand for the equivalent cost per unit of power loss and the annual capacitor installation cost, respectively. The total power losses of the network is presented by $P_{T,loss}$. The amount of reactive power which compensated in ith bus defined by Q^{C}_{i}.

$$X = [T_1, T_2, ..., T_{N_b}, Q_1^c, Q_2^c, ..., Q_{N_b}^c]_{1\times(2N_b)} \tag{3}$$

Where Ti that corresponds to the two discrete values; 0 and 1. It takes zero value when ith bus is not a suitable for compensation otherwise takes value 1 as a result kVAr (the reactive power) can applied to the ith bus.

Traditional methods for the feeder power losses is determined through (4) [27].

$$f_2(X) = p_{loss}(X) = \sum_{i=1}^{N_{br}} R_i \times |I_i|^2 \tag{4}$$

In which Ri and Ii are resistance and actual current of the ith branch.

The magnitude of voltage deviation is another objective function which need to follow (5) [28].

$$f_3(X) = d_{volt}(X) = \max\left\{ |1 - V_{\min}|, |1 - V_{\max}| \right\} \tag{5}$$

2.2 Practical constrains

All limitation and operational constraints which must preserved through analysis for meet the practical conditions are determined and described as follow.

2.2.1 Capacitor sizes

The size of capacitors is discrete value related to the smallest capacitor size. From economic point of view, the large capacitors size imposes lower cost than the small size one. Consequently, the available capacitor bank size can be described in (6).

$$Q_{M\,ax}{}^{c} = L \times Q_0^{c} \tag{6}$$

In which Q^c_0 denote the smallest size of the capacitors in kVAr that multiply an integer (L). The available capacitor sizes and the correspond cost are summarized in Table 1.

Table 1. The available capacitor sizes and cost

i	1	2	3	4
Qc (kVAr)	150	300	450	600
$/ kVAr	0.500	0.350	0.253	0.220
i	5	6	7	8
Qc (kVAr)	750	900	1050	1200
$/ kVAr	0.276	0.183	0.228	0.170
i	9	10	11	12
Qc (kVAr)	1350	1500	1650	1800
$/ kVAr	0.207	0.201	0.193	0.187
i	13	14	15	16
Qc (kVAr)	1950	2100	2250	2400
$/ kVAr	0.211	0.176	0.197	0.170
i	17	18	19	20
Qc (kVAr)	2550	2700	2850	3000
$/ kVAr	0.189	0.187	0.183	0.180
i	21	22	23	24
Qc (kVAr)	3150	3300	3450	3600
$/ kVAr	0.195	0.174	0.188	0.170
i	25	26	27	
Qc (kVAr)	3750	3900	4050	
$/ kVAr	0.183	0.182	0.179	

2.2.2 Distribution Line Capacity

The power flow over the feeders should follow the below constraint to guarantee the safe flow among buses.

$$\left| P_{ij}^{Line} \right| < P_{ij,\max}^{Line} \tag{7}$$

Where $|P^{Line}ij|$ as an absolute power flow must be less than maximum power flow over the distribution line between two nodes of *i* and *j*.

2.2.3 Bus voltage limitation

There is a limitation for the voltage level of ith bus to maintain the operating voltage range between the minimum and maximum constraints (*Vmin*, *Vmax*) as is formulated in (8).

$$V_{\min} \le V_i \le V_{\max} \tag{8}$$

3. 2m PEM STOCHASTIC ANALYSIS

The optimal capacitor allocation problem for radial distribution networks incorporates several uncertain parameters which an appropriate probabilistic analyzer should utilize to consider intrinsic random characteristic of the problem variables. From the viewpoint of technical mathematics, the 2m PEM account as a more accurate stochastic method compare to analytical techniques. Despite the simulation based methods (Monte Carlo Simulation) the proposed stochastic analyzer based on 2m PEM show less processing time for converging to optimum point. The main characteristics behind approximate methods such as 2m PEM would be its appropriate capability for handling the uncertainties in the search space. As a positive point of 2m PEM can points out to less need to data (the mean, variance, skewness, and etc.) from the probability function of the uncertain variables. In order to describe the mathematics rule of the 2m PEM, it is simpler to formulate the deterministic power flow function as a non-linear function as shown in (9).

$$S = F(z) \tag{9}$$

In which *z* is defined as an input vector related to load, branch data, and the topology of network. Correspondingly, *S* output vector consists the information about line power flow, and voltage profile is perspective of the uncertainty in the input vector *z*. In other word, 2*m* PEM runs power flow function to approximate the output vector *S* moments. According Eq. (9) a few number of moments of *z* are needed in which for l^{th} input variable of *z* vector, a f_{zl} (probability distribution function) is defined as shown in (10) **Error! Reference source not found.**. The initial values of probability point's is selected by f_{zl} matching with two points through mean, variance, and skewness coefficient according 2*m* PEM.

$$S = F(\mu_{z1}, \mu_{z2}, ... z_{l,k}, ..., \mu_{zm}); k = 1,2 \tag{10}$$

Where $z_{l,k}$ is kth location point of the random variable z_l. Two variables $z_{l,1}$ and $z_{l,2}$ stand for two random variables location of z_l which is defined in (11). Also, the standard location of the random variables z_l ($\zeta_{l,k}$) is selected as follow:

$$z_{l,k} = \mu_{z_l} + \xi_{l,k}.\sigma_{z_l}; \quad k = 1,2 \tag{11}$$

$$\xi_{l,k} = \frac{\lambda_{l,3}}{2} + (-1)^{3-k}\sqrt{m - (\lambda_{l,3}^2/2)^2}, \quad k = 1,2 \tag{12}$$

Where μ_{zj} and σ_{zl} defined as mean value of the j^{th} random variable and standard deviation of the random variable z_l , respectively. Also, $\lambda_{l,3}$ stand for the skewness coefficient. The 2*m* PEM transferred the space variables in the locations $z_{l,1}$ and $z_{l,2}$ to the output variables $S_{l,1}$ and $S_{l,2}$ by utilizing weight parameters. In this case, two weight parameters are used to map $z_{l,1}$ and $z_{l,2}$ on the output data set. Finally, the third central moment ($\lambda_{l,3}$) is defined based on expected operator (*E*) as follow:

$$\lambda_{l,3} = \frac{E\left[\left(z_l - \mu_{z_l}\right)^3\right]}{\left(\sigma_{z_l}\right)^3} \tag{13}$$

Accordingly, the standard deviation value (σ) of S_i is indicated by variance of output vector (var) as below.

$$\sigma = \sqrt{\mathrm{var}(S_i)} = \sqrt{E(S_i^2) - \left[E(S_i)\right]^2} \tag{14}$$

4. MULTI-OBJECTIVE OPTIMIZATION ALGORITHM

As stated before, the capacitor allocation optimization problem is a type of discrete, complex, and nonlinear integer programming associated with multiple objective functions which some of these fitness functions correlated to each other. Thus, the powerful optimization tool will require to cope with all non-linearity and Pareto problems. This paper proposed a self-adaptive modified harmony search (MHS) algorithm to address the deficiencies of the previous methods for handling the conflicting targets of the problem. Mathematically, the concerned multi-objective optimization problem is formulated in (15) incorporate to the constraints [30].

$$\begin{aligned} &\min F = [f_1(X), f_2(X), ..., f_n(X)]^T; \\ &h_i(X) = 0 \quad i = 1,2,...,N_{eq} \\ &g_i(X) < 0 \quad i = 1,2,...,N_{ueq} \end{aligned} \tag{15}$$

Moreover, two main below conditions must be satisfied in terms of multi-objective minimization problem in which the X1 solution determine as a dominant solution if both (16) and (17) are satisfied as below.

$$\forall j \in \{1,2,...,n\}, f_j(X_1) \le f_j(X_2) \tag{16}$$

$$\exists k \in \{1,2,...,n\}, f_k(X_1) < f_k(X_2) \tag{17}$$

In the case of not satisfying any of the two above conditions, then this results not selecting the X1 solution as dominated one compared to X2 so it counts as non-dominated solutions.

Generally, in a given fuzzy system, linguistic variables are defined to model the system behavior as word or sentence compared to the mathematical models which is defined as numerical values. For a given engineering problem, linguistic variables might be different depends upon the area of problem. For example, fuzzy logic controller is one of the most well-known application of fuzzy logic in which linguistic variables can be defined as error and change of error [31]. The complete set of optimization process data is stored in repository which its size needs to be controlled by fuzzy-based clustering technique. There are several fuzzy membership functions available in the literature namely linear, Gaussian, singleton, triangular, trapezoidal. Here, fuzzy membership function offers the capability to maintain all value associate with the concerned objective functions in the same base in a way that repository size keeps constant. In this case, a fuzzy membership function for each one of the objectives functions is dedicated as formulated in (18). For each one of the Pareto optimal solution sorted in repository, a normalized membership function is evaluated in (19) based on available decision options.

$$MF_i(X)=\begin{cases}1 & for\ \ f_i(X)\le f_i^{\min}\\ 0 & for\ \ f_i(X)\ge f_i^{\max}\\ \dfrac{f_i^{\max}-f_i(X)}{f_i^{\max}-f_i^{\min}} & f_i^{\min}\le f_i(X)\le f_i^{\max}\end{cases} \tag{18}$$

$$N_{MF}(j)=\frac{\sum_{i=1}^{n}\Delta_i\times MF_i(X_j)}{\sum_{j=1}^{Np}\sum_{i=1}^{n}\Delta_i\times MF_i(X_j)} \tag{19}$$

Where $f_i^{\min}$, $f_i^{\max}$ are defined as minimum and maximum boundaries of the i[th] objective function (fi(X)). In (19), $MF_i(X)$ is described as fuzzy membership function for the i[th] objective function. The fuzzy membership function determines as an adaptive decision making criteria which can change based on available decision candidate. All non-dominated solutions are evaluated by NMF and at the end the best non-dominated solution will save in the repository.

5. SELF-ADAPTIVE MODIFIED HARMONY SEARCH (SAMHS)

5.1 Original harmony search algorithm

Harmony search mimics by process of musician search for finding an ideal harmony. It has a tight relationship with the composing process of each harmony. Each harmony is perspective of one solution for concerning optimization problem. The main steps of the harmony search (HS) algorithm are summarized as follow **Error! Reference source not found.**:

Step 1- Parameters initialization: The setting parameters of the algorithm such as decision variables, upper/lower boundaries of each variable, harmony memory (HM) size, harmony memory consideration rate (HMCR), pitch adjusting rate (PAR), and bandwidth vector (BW) are initialized in this step.

Step 2- Harmony memory (HM) initialization: A random matrix for HM is generated in which each component of this matrix is initialized using uniform random number between upper/lower boundaries of the decision variables.

Step 3- New harmony improvisation: Generally, there are three main rules which are utilized for new harmony improvisation. Each note in HM is played by a musician to get to the highest harmony based on three subsets that are listed in below:

- Harmony memory consideration: Each component of the new harmony vector is selected from stored corresponding components in the HM randomly. This selection is performed based on HMCR formulated in (20) as the selecting probability which is determined between 0 and 1.

$$x_{kh}^{new}=\begin{cases}x_{kh}^{HM} & rand \prec HMCR\\ x_{kh}^{rand} & otherwise.\end{cases} \tag{20}$$

- Pitch adjustment: The selected components from HM modify depends upon PAR and BW which stand for the probability of a selected for modification component from HM and distance bandwidth, respectively. Wherein the BW parameter as the bandwidth is updated in each iteration (i) as follows:

$$x_{kh}^{new}=\begin{cases}x_{kh}^{HM}\pm rand\times BW & ;\ \ rand<PAR\\ x_{kh}^{rand} & ;\ \ Otherwise\end{cases} \tag{21}$$

$$BW(i)=BW_{\max}\times e^{\rho i} \tag{22}$$

$$\rho=Ln(\frac{BW_{\min}}{BW_{\max}})\times NI^{-1} \tag{23}$$

In this step, each new produced note checks to assure that it does not need more tune. In other word, after performing step 1 and initializing parameters such *HM*, *BW*, and *PAR* an internal updating step need to adopt *PAR* varying in regard to search capability improvement by total number of improvisation stages (*NI*). Thus, updating *PAR* value dynamically is computed in (24).

$$PAR(i)=PAR_{\min}+\frac{i}{NI}\times(PAR_{\max}-PAR_{\min}) \tag{24}$$

Where the pitch adjustment rate is defined as $PAR(i)$ in generation I at population. Also, PAR_{max} and PAR_{min} stand for maximum and mini-mum tolerance of adjustment rate.

- Random generation: A random selection among each not selected components from memory consideration is performed between upper/lower bounds. As a result, the generation probability of random component would be (1-HMCR).
- Harmony memory update: The new improved harmony is assessed based on the objective function in a way that if the objective function value of new harmony vector is better than its peer of worst harmony in the HM then the current value of harmony is replaced with the new one. After maximum possible number of improvisation is reached, the process stopped and the best harmony (solution) is store in memory. If the stopping criterion does not meet step 3 and 4 are repeated.

5.2 Self-adaptive Sub-modification technique

In the original HS algorithm, setting small PAR value along with large value of BW increase computational burden needed for exploring the optimal solution as a result deteriorate performance of optimization tool. Therefore, the original HS method needs some modification to improve its performance for multi-objective problem particularly. In this paper a modification stages are explained to improve the improvisation steps beside increase the optimization algorithm convergence capability depends upon the adaptive modified HS algorithm. The proposed self-adaptive technique is divided into two sub-modifications which each one can be adaptively selected according to their necessities during the optimization process.

5.2.1 Sub-modification1:

As stated above, in each iteration (i) of the original HS a four-step procedure is applied to create and update new harmony. Clearly, this procedure can be merge with genetic operators, i.e., crossover and mutation in order to improve HS performance. In the original HS, the new produced notes are generated as formulated in (25).

$$X_{Mut,1} = X_{n1} + \Im_1 \times (X_{n2} - X_{n3}) \tag{25}$$

Where three notes, X_{n1}, X_{n2}, and X_{n3} are selected from population in random such that $n_1 \neq n_2 \neq n_3$. By using the mutation operator. Then, by using crossover operator three new notes are generated as follow:

$$\overline{X_{n,2}} = [x_{nn2,1}, x_{nn2,2}, \ldots, x_{nn2,2N_b})]; \quad x_{nn1,2N_b})] \tag{26}$$

$$x_{nn1,i} = \begin{cases} x_{mut,i}, & if \ \ \gamma_1 \le \gamma_2 \\ x_{note,i}, & \text{otherwise} \end{cases}$$

$$x_{nn2,i} = \begin{cases} x_{mut,i}, & if \ \ \gamma_2 \le \gamma_3 \\ x_i, & \text{otherwise} \end{cases}$$

Now the best note is selected based on modified step for the new improved harmony. The modification step will increase the diversity of the harmony population, sufficiently.

5.2.2 Sub-modification2:

In the second modification step, value of $\Im$ update based on (27). The adaptive formulation of $\Im$ update is described after several consecutive running process.

$$\Im^{k+1} = (1/3Iter)^{1/Iter} \Im^k \tag{27}$$

Each sub-modification steps will determine by specific probability to be chosen correctly. First, both sub-modifications determine with the same probability (P_θ= 0.33; θ=1,2). Then, accumulator (**A**) is defined to recognize the probability of each sub-modification which is assigned initially zero. Next, X_1 and X_{Npop} as the best and worst solutions in the harmony population will sort according to their fitness function in the descending order. Finally, for each harmony will be allocated a weighting factor (W) as shown in (28).

$$W_j = \frac{\log(N_{pop} - j + 1)}{\log(1) + \ldots + \log(N_{pop})} \quad j = 1, \ldots, N_{pop} \tag{28}$$

Now, all the accumulators are updated through (29).

$$A_\theta = A_\theta + \frac{Wgt_l}{n_{Mod_\theta}} \quad l = 1, \ldots, n_{Mod_\theta} \tag{29}$$

In the above equation, $\eta_{Mod\theta}$ shows the number of notes which have chosen θ^{th} sub-modification step. In addition, accumulator value will determine the probability of sub-modification methods in each iteration (*Iter*) as follow:

$$P_\theta = (1 - \nabla) \times P_\theta + \nabla \times \frac{A_\theta}{Iter}, \ (\theta = 1,2,3) \tag{30}$$

Where ∇ stand for the learning speed which equals ω=0.142, experimentally. Finally, the normalized probability of each modification method is evaluated in (31).

$$P_\theta = P_\theta / (\sum_{\theta=1}^{3} P_\theta) \tag{31}$$

For each iteration, each harmony will choose θ^{th} sub-modification methods according their probabilities.

6. SOLUTION PROCEDURE

The sequence of steps for applying SAMHSO algorithm to minimize/maximize optimal objective functions of the problem, are given through flowchart below:

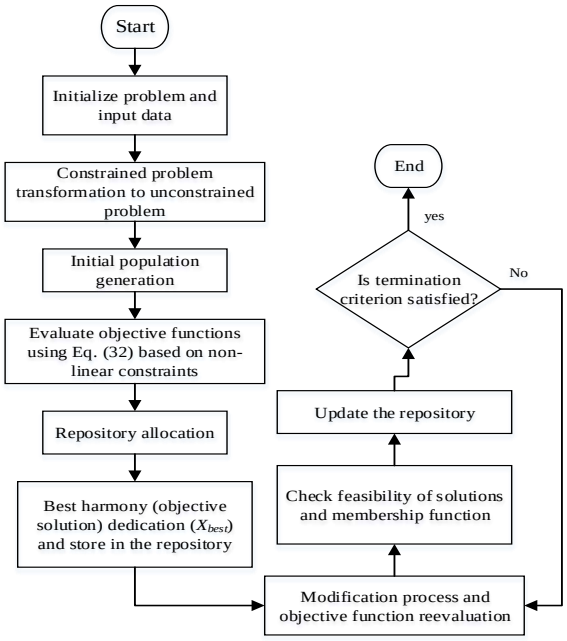


Figure. 1 Block diagram of SAMHSO algorithm by the proposed method

The proposed optimization method with self-adaptive modification techniques is implemented for the multi-objective capacitor allocation optimization problem. First, all economic data, optimization constraints, termination criterion, and input date of optimization tool, and objective functions are defined then constrained problem is transformed to unconstrained.

In the next step, the non-dominated solution result of the first phase of objective function evaluation are stored in the repository and recheck its size. Then, the best solutions are chosen depends upon their constraints. Next, the population is improved by the sub-modification steps to improve intensification and diversification of optimization process. After modifying the population, the objective functions reevaluate and feasibility of them are checked. If the best individual dominates the i^{th} harmony, the harmony will be replaced by that individual else it will remain in its position. Thus, the repository is updated based on the new places and then termination criterion is checked. If criteria satisfied finish the algorithm, otherwise, go to modification step and repeat the process.

7. SIMULATION

In this section, the concerned optimization problem is solved utilizing a notional distribution network **Error! Reference source not found.**. Fig. 2 shows the single-line diagram of the test system which is investigated as the case study. The case study is IEEE 9-bus test system with radial structure supply one feeder with the line voltage 23 kV.

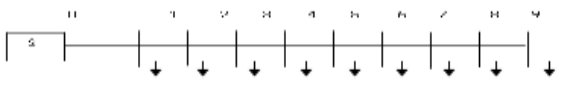


Figure. 2 IEEE 9-bus distribution test system

The network data details of case study can be found in **Error! Reference source not found.** regard feeder topology and load characteristics. The nominal active and reactive load values are 12368 kW and 4186 kVAr, respectively. The simulation results divided to two phases, deterministic and stochastic analysis. In the first part, the proposed SAMHSO algorithm is applied to solve capacitor allocation for each of three objective functions (loss, and voltage deviation and annual price). Fig. 3 are shown to show the investigated objective functions for 9 buses system. Also, Table.2 depicted to obtain the capacitor size and corresponding locations of the test system based on SAMHSO algorithm.

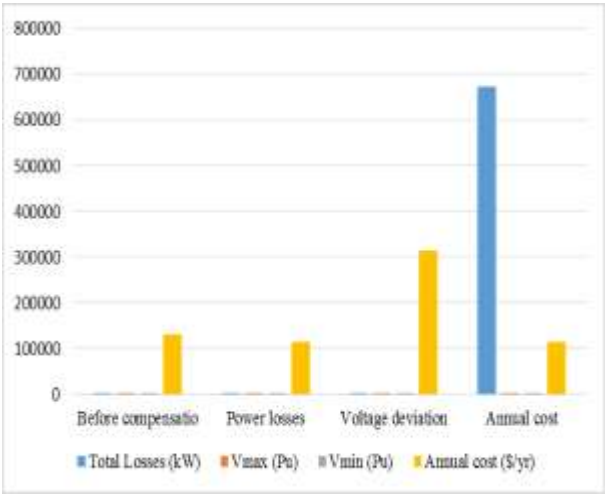


Figure. 3. Deterministic analysis of single objective function depends upon the proposed SAMHSO

As it can be seen from Fig.3, the objective functions' values effectively decreased by using the proposed SAMHSO algorithm compare to the case before compensation. The annual price and active power losses are shown same behavior in the single optimization process. Despite these both cases, the voltage deviation is high compare to the base case without compensation. By defining the voltage deviation as the single objective function can attain the minimum value of voltage deviation in range of 0.00501(Pu) which is a satisfying value. However, the power losses and annual price are increased. Therefore, all objective functions must be improving effectively in the case of single objective optimization problem.

Table 2. Capacitor location and size through deterministic analysis

	Capacitor location (bus number)							
Bus No.	1	2	3	4	5	6	7	9
Optimum Power losses	150	4050	1350	2100	900	150	300	300
Bus No.	5	7	8	9				
Optimum Voltage deviation	600	4050	1950	3450				
Bus No.	2	3	4	5	7	9		
Optimum Annual cost	3300	900	2100	900	450	300		

For comparing the robustness of the proposed algorithm with the well-known optimization algorithms **Error! Reference source not found.** which are recently investigated by the researchers, Fig.4, 5, and 6 are depicted for capacitor allocation problem in the distribution network. The multi-objective problem by using the SAMHSO algorithm is applied to case study by the bus voltage limits in the range of [0.9,1.1] Pu.

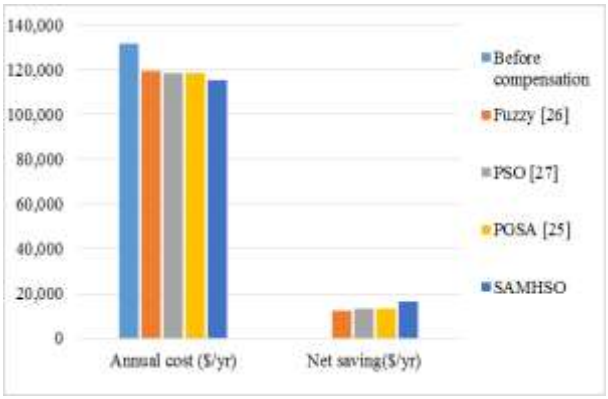
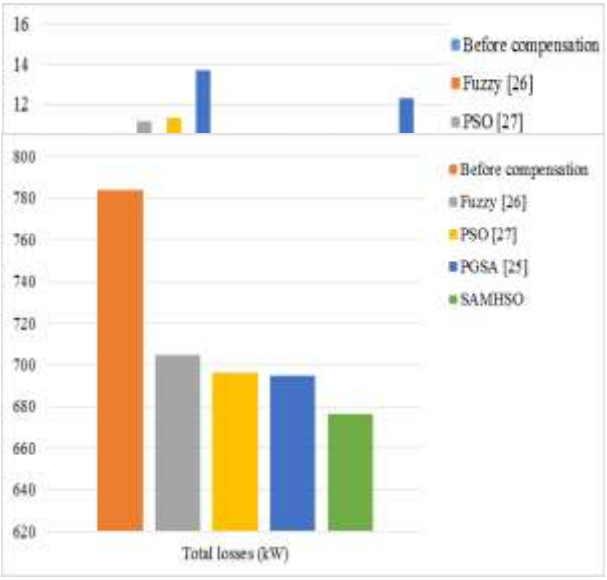


Figure. 4. Deterministic analysis of single objective function depends upon the proposed SAMHSO

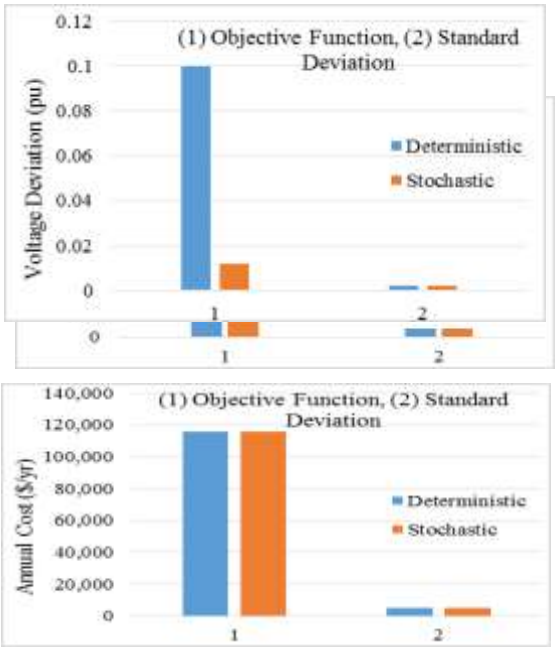
Figure. 5. Comparison of deterministic analysis of capacitor allocation by different algorithms for loss reduction and saving percent

Figure. 6. Comparison of deterministic analysis of capacitor allocation by different algorithms for Total loss (kW)

As it can be seen from above figures (4, 5, and 6), the proposed optimization method (SAMHSO) shows better performance in finding the pareto solution than the other well-known methods **Error! Reference source not found.**. Thus, the performance of SAMHSO algorithm should assay in stochastic analysis. In order to involve the probabilistic analyzer to the capacitor allocation problem, the *2m*PEM is consider into the problem. Result are shown in Fig.7 describe the best optimum values for multi-objective functions for IEEE 9 buses case study. For the sake of better and precise comparison, the results compared with the deterministic framework.



(a)



(b)

(c)

Figure. 7. Comparison of deterministic and stochastic analysis of capacitor allocation of multi-objective optimization

The stochastic analysis cost is shown into incremental growth of the optimal values of the objective functions. In fact, the uncertain parameters impose some non-linearity into the problem which by considering the uncertainties the optimal values are increased. However, considering the uncertainties offers more realistic analysis for the real applications. As it can be seen from results reported in Fig.7, the proposed probabilistic method has diminished the standard deviation values of the multi-objective functions. It is noteworthy to note that the proposed approach with probabilistic analysis still shows better performance compared to the other common used methods.

8. CONCLUSION

Like several heuristics methods such as PSO, GA, and GSA for solving the capacitor allocation into distribution network a need for a novel optimization algorithm would be necessary which be capable to cope with high volume of non-linearity in the problem. In this paper an efficient and precise stochastic framework is considered depends upon SAMHSO algorithm associated with 2m PEM. The proposed self-adaptive sub-modification techniques is consisted of two different modification method. Optimal capacitor allocation is solved

by SAMHSO algorithm and its capability in solving non-convex problem will increase the convergence ability of method. The proposed approach has been tested on IEEE 9 buses case study with superior results regarding the fast convergence and benefits of networks compared to most common used methods. Moreover, effectiveness and feasibility of the proposed method was shown through multi-objective functions optimization problem.

9. REFERENCES

[1] Das, D. "Optimal placement of capacitors in radial distribution system using a Fuzzy-GA method." International Journal of Electrical Power & Energy Systems 30.6 (2008): 361-367.

[2] B. Papari, C. S. Edrington, T. Vu, “Stochastic Operation of Interconnected Microgrids”, IEEE Power and Energy Society (PES), 2017, Chicago, USA.

[3] Khakpour, A., D. Asemani, and B. Papari. "Adaptive Noise Cancellation based Architecture for Correction of Gain and Offset Mismatch Errors in Time-Interleaved ADC." Iranian Journal of Science and Technology. Transactions of Electrical Engineering 38.E2 (2014): 149.

[4] C. F Chang, Reconfiguration and Capacitor Placement for Loss Reduction of Distribution Systems by Ant Colony Search Algorithm, IEEE Trans. Power. Syst, 23(4) (2008) 1747-1755.

[5] Papari, B., D. Asemani, and A. Khakpour. "A wide-band time-interleaved A/D converter for cognitive radio application with adaptive offset correction." Wireless Advanced (WiAd), 2011. IEEE, 2011.

[6] R. Baldick, F.F. Wu, Efficient integer optimization algorithms for optimal coordination of capacitors and regulators, IEEE Trans. Power Syst. 5 (1990) 805–812.

[7] Vu, T. V., Perkins, D., Papari, B., Vahedi, H., & Edrington, C. S. (2017, June). Distributed adaptive control design for cluster of converters in DC distribution systems. In DC Microgrids (ICDCM), 2017 IEEE Second International Conference on (pp. 197-201). IEEE.

[8] Jabr, R. A. "Optimal placement of capacitors in a radial network using conic and mixed integer linear programming." Electric Power Systems Research 78.6 (2008): 941-948.

[9] Vu, T., Gonsoulin, D., Perkins, D., Papari B., Vahedi, H., Edrington, C. S. (2017, August). Distributed Control Imple-mentation for Zonal MVDC Ship Power Systems. In Electric Ship Technologies Symposium (ESTS), 2017 IEEE (pp. 14-18).

[10] Olamaei, J., M. Moradi, and T. Kaboodi. "A new adaptive modified firefly algorithm to solve optimal capacitor placement problem." Electrical Power Distribution Networks (EPDC), 2013 18th Conference on. IEEE, 2013.

[11] Al-Kandari, A.M., Soliman, S.A. and El-Hawary, M.E., 2004. Fuzzy short-term electric load forecasting. International Journal of Electrical Power & Energy Systems, 26(2), pp.111-122.

[12] G. Sahamijoo, O. Avatefipour, M. R. S. Nasrabad, M. Taghavi, F. Piltan, Research on minimum intelligent unit for flexible robot,International Journal of Advanced Science and Technology, Vol. 80,No. 6, pp. 79-104, 2015.

[13] Mokhtar, M., Piltan, F., Mirshekari, M., Khalilian, A., & Avatefipour, O. (2014). Design minimum rule-base fuzzy inference nonlinear controller for second order nonlinear system. International Journal of Intelligent Systems and Applications, 6(7), 79.

[14] Khalilian, A, Piltan, F, Avatefipour, O, Safaei MR, Sahamijoo, G. Design New Online Tuning Intelligent Chattering Free Fuzzy Compensator, International Journal of Intelligent Systems and Applications(IJISA), vol.6, no.9, pp.75-86, 2014. DOI: 10.5815/ijisa.2014.09.10

[15] Shahcheraghi, A., Piltan, F., Mokhtar, M., Avatefipour, O., & Khalilian, A. (2014). Design a Novel SISO Off-line Tuning of Modified PID Fuzzy Sliding Mode Controller. International Journal of Information Technology and Computer Science (IJITCS), 6(2), 72.

[16] Park, J.H., Seo, S.J. and Park, G.T., 2003. Robust adaptive fuzzy controller for nonlinear system using estimation of bounds for approximation errors. Fuzzy Sets and Systems, 133(1), pp.19-36.

[17] Avatefipour, O., Piltan, F., Nasrabad, M. R. S., Sahamijoo, G., & Khalilian, A. (2014). Design New Robust Self Tuning Fuzzy Backstopping Methodology. International Journal of Information Engineering and Electronic Business, 6(1), 49.

[18] B Papari, CS Edrington, F Kavousi-Fard, “An Effective Fuzzy Feature Selection and Prediction Method for Modeling Tidal Current: A Case of Persian Gulf”, IEEE Trans. Geoscience and Remote Sensing, vol. 99, 2017, pp. 1-6.

[19] Bhattacharya S.K, S.K. Goswami , A new fuzzy based solution of the capacitor placement problem in radial distribution system, Elzevier Expert Systems with Applications 36 (2009) 4207-4212.

[20] Abdelaziz, A. Y., E. S. Ali, and SM Abd Elazim. "Optimal sizing and locations of capacitors in radial distribution systems via flower pollination optimization algorithm and power loss index." Engineering Science and Technology, an International Journal 19.1 (2016): 610-618.

[21] Tabatabaei, S. M., and B. Vahidi. "Bacterial foraging solution based fuzzy logic decision for optimal capacitor allocation in radial distribution system." Electric Power Systems Research 81.4 (2011): 1045-1050.

[22] Papari. B., Asemany, D., Khakpour, A., “An Adaptive Mismatch Error Cancellation Architecture for Time-Interleaved A/D Converters.” International a journal of Electronics and Communication Technology (IJECT); vol. 4, Spl. 5, 2013, pp. 9-13.

[23] Grainger, John J., and S. H. Lee. "Optimum size and location of shunt capacitors for reduction of losses on distribution feeders." IEEE Transactions on Power Apparatus and Systems 3 (1981): 1105-1118.

[24] Huang, Shyh-Jier. "An immune-based optimization method to capacitor placement in a radial distribution system." IEEE Transactions on Power Delivery 15.2 (2000): 744-749.

[25] Lee, Kang Seok, and Zong Woo Geem. "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice."

Computer methods in applied mechanics and engineering 194.36 (2005): 3902-3933.

[26] R. Srinivasa Rao, S. V. L. Narasimham, Optimal Capacitor Placement in a Radial Distribution System using Plant Growth Simulation Algorithm, World Academy of Science, Engineering and Technology 45 (2008) 715-722.

[27] C. T. Su, C. C. Tsai, A new fuzzy reasoning approach to optimum capacitor allocation for primary distribution systems, Proc IEEE Indus. Tech.Conf. (1996) 237–41.

[28] B Papari, CS Edrington, I bhattacharya, G Radman, "Effective Energy Management of Hybrid AC-DC Microgrids with Storage Devices", IEEE Trans. Smart Grid, vol. 99, 2017, pp. 1-6.

[29] Yu, Xin-mei, Xin-yin Xiong, and Yao-wu Wu. "A PSO-based approach to optimal capacitor placement with harmonic distortion consideration." Electric Power Systems Research 71.1 (2004): 27-33.

[30] El-Fergany, Attia A., and Almoataz Y. Abdelaziz. "Artificial bee colony algorithm to allocate fixed and switched static shunt capacitors in radial distribution networks." Electric Power Components and Systems 42.5 (2014): 427-438.

[31] Papari, B., C. S. Edrington, T. V. Vu, and F. Diaz-Franco. "A heuristic method for optimal energy management of DC microgrid." In DC Microgrids (ICDCM), 2017 IEEE Second International Conference on, pp. 337-343. IEEE, 2017.

[32] Olamaei, J., M. Moradi, and T. Kaboodi. "A new adaptive modified firefly algorithm to solve optimal capacitor placement problem." Electrical Power Distribution Networks (EPDC), 2013 18th Conference on. IEEE, 2013.

[33] Anderson, Adam L., C. Brett Witherspoon, and B. Papari. "Spectrum recognition in large-scale cognitive radio networks with spectral data mining." Proceedings of WORLDCOMP (2014).

[34] Khalilian, A., Sahamijoo, G., Avatefipour, O., Piltan, F., & Nasrabad, M. R. S. (2014). Design high efficiency-minimum rule base PID like fuzzy computed torque controller. International Journal of Information Technology and Computer Science (IJITCS), 6(7), 77.

[35] Lin, Chi-Ming, and Mitsuo Gen. "Multi-criteria human resource allocation for solving multistage combinatorial optimization problems using multiobjective hybrid genetic algorithm." Expert Systems with Applications 34.4 (2008): 2480-2490.

[36] T. Niknam, A. Kavousifard, S. Tabatabaei, J. Aghaei, Optimal operation management of fuel cell/wind/photovoltaic power sources connected to distribution networks, J. Power Sources 196 (2011) 8881– 8896.

International Journal of Computer Applications Technology and Research
Volume 6–Issue 12, 493-497, 2017, ISSN:-2319–8656

Energy Optimization using Cloud Offloading Algorithm

Jyothi T
Assistant Professor
Department of ISE
GSSSIETW, India

***Abstract*—Computational offloading is an effective method to address the limited battery power of a mobile device, by executing some components of a mobile application in the cloud. In this paper, a novel offloading algorithm called 'Dynamic Programming with Hamming Distance Termination' (denoted DPH) is presented. The algorithm uses randomization and a hamming distance termination criterion to find a nearly optimal offloading solution quickly. The algorithm will offload as many tasks as possible to the cloud when the network transmission bandwidth is high, thereby improving the total execution time of all tasks and minimizing the energy use of the mobile device. Furthermore, the algorithm is extensible to handle larger offloading problems without a loss of computational efficiency.**

Keywords —***Mobile Cloud Computing; Dynamic Programming; Computational Offloading; Randomization; Hamming Distance; Energy-Efficiency***

I. INTRODUCTION

Computation offloading is a method where some of the computational tasks of a mobile application can be offloaded to run on remote servers in the cloud, in order to save energy [1] [2]. However, the problem of partitioning the application tasks for offloading is NP-complete in general. The main goal of the offloading algorithm is to minimize the overall energy used by the mobile application, while meeting an execution time constraint.

A task to be offloaded must be transmitted over a wireless access network, and the time-varying wireless transmission bandwidth must be considered. An adaptive offloading algorithm can determine the offloading decisions dynamically according to a changing wireless environment.

Reference [3] presented a system that enables energy-aware offloading of mobile tasks to the

cloud called MAUI. Further improvements were proposed in CloneCloud [4] and Thinkair [5]. In all cases, the partitioning problem results in an integer programming problem which cannot be solved efficiently. A Dynamic Programming (DP) algorithm was proposed in [6], where a two dimensional DP table was used. However, this scheme did not consider an execution time constraint when computing the offloading decisions, although this time constraint is an important issue for many interactive applications . Furthermore, a backtracking algorithm was needed to find the final decisions, which was time consuming.

Reference [7] provided a dynamic programming approach which builds a three-dimensional programming table and requires pseudo polynomial time complexity [7]. However, it doesn't consider the energy consumed in the mobile device which is an important criteria for mobile devices.

In this paper, an innovative dynamic programming algorithm called DPH is proposed. Dynamic programming is an optimization approach that transforms a complex problem into a sequence of simpler problems. The DPH algorithm introduces randomization, i.e., we generate random bit strings of 0s and 1s periodically and utilize sub-strings when they improve the solution (similar to genetic optimization). We also fill the dynamic programming table in a creative way to avoid the extra computation for common sub-strings.

The algorithm can find a nearly optimal solution after several iterations. It uses a Hamming distance criterion to terminate the search to obtain the final decision quickly. The hamming distance termination criterion is met when a given fraction of tasks are uploaded. The

final solution depends upon the wireless network transmission bandwidth and the computational power of the CAP and cloud servers.

The remainder of the paper is organized as follow: Section II provides the system model and problem formulation. Section III presents the proposed algorithm which is based on the dynamic programming table. The paper concludes in section IV.

II. SYSTEM MODEL AND PROBLEM FORMULATION

Consider an application consisting of some un-offloadable (i.e., local) tasks and N offloadable tasks. Normally, local tasks include those that directly handle user interaction, access local I/O devices or access specific information on the mobile device. Therefore, local tasks must be locally processed by the mobile user. We can merge all the local tasks into one task [13]. In [14], an example of a face recognition problem which is composed of eight offloadable tasks and one local task is presented.

A. Network Model

We consider a handheld mobile device with N independent tasks that can be executed locally or transferred to cloud for execution as shown in Fig. 1. We assume that a WiFi wireless network is available for the mobile device, but the network transmission bandwidth can change dynamically. Typically,wireless interference and network congestion will dynamically change the network transmission bandwidth. The mobile device needs to decide whether each task should be processed locally or offloaded, according to the current wireless network transmission bandwidth. The time taken to transfer a task between a mobile device and the cloud through a wireless link is an important issue since a total execution time constraint for all tasks exists. Therefore, the dynamic programming algorithm must consider the current wireless network bandwidth when computing a decision.

For task i, let Mi ∑ {0, 1} be an execution indicator variable. Let Mi = 1 if task i is executed at the mobile device and 0 otherwise. If it is executed locally, the energy consumption is Eli. Eri is the energy consumption of the mobile device when the task i is executed on the cloud, and Eti is the energy-cost of transmitting task i to the cloud server.

Variable Tli is the local execution time to process task i, and Tri is the remote execution time when task i is executed in remote cloud server. Variable Tti is the transmission time to transfer task i to the cloud server.

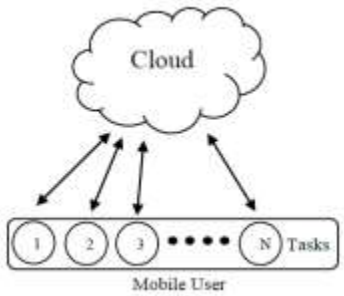


Fig 1: Network Model

It is clear that the transmission energy used to upload each task will depend on the network transmission bandwidth. Therefore, changes in wireless network bandwidth will affect the offloading decision. For example, if we assume that transmission time of each task is equal to the size of each task divided by the network transmission rate, then any variation in the transmission rate will affect the final decision of whether to offload this task or not.

The energy consumption function and its corresponding execution time are defined in (1) and (2):

$$E = \sum_{i \in N} (M_i El_i + (1 - M_i) Er_i + (1 - M_i) Et_i) \quad (1)$$

$$T = \sum_{i \in N} (M_i Tl_i + (1 - M_i) Tr_i + (1 - M_i) Tt_i) \quad (2)$$

The execution time T of all tasks must satisfy the following condition, where Tconstraint is the execution time requirement

$$T \leq T_{cnostraint}$$

For simplicity, we use M = [M1, M2, MN] to denote a vector of binary offloading decisions. The problem that we want to solve is as follow:

$$\min \quad E$$

$$\text{Subject to:} \quad T \le T_{constraint}$$

The number of combinations of binary values Mi to search for the optimal solution grows exponentially with the number of tasks. Our goal is to determine which tasks should be offloaded to the cloud server to minimize energy while meeting a mobile application's execution time constraint.

III PROPOSED ALGORITHM BASED ON THE DYNAMIC PROGRAMING TABLE

A. Innovative Way of Filling the Table

The proposed algorithm is called 'Dynamic Programming with Hamming Distance Termination' (DPH). In this scheme, we use an N*N table to store the bit-streams that show which tasks should be offloaded (where N is the number of tasks). For the first step, a random bit stream is generated that determines a first solution.

This stream is assigned to the table such that 1s are assigned to the next horizontal cell, and the 0s are assigned to the next vertical cell. If the first bit of the stream is 1, the starting cell is (1, 2) and if the first bit of the stream is 0, the starting cell is (2, 1).

This approach will avoid extra computations for common bit strings. A 2D 8*8 table is shown in table I. To clarify, assume that N = 8 and the first random stream is 11100110 (black numbers) or 00110110 (red numbers), (2 examples are given). Assume that the second random bit stream in each case is 11000111.

The starting cell of the second stream is (1, 2) since the first bit is 1. By following the aforementioned rules to fill the table, the resulting green stream is shown in table I.

Table I. How to fill the table for 2 examples.

	1	1	1				
0			0				
0	1	1	0	1	1		
		0	1	1	0		
				0			

	1	1	1				
0		0	0				
0	1	1/0	0	1	1		
		0	1	1	0/1		
				0			

Whenever a bit stream is generated randomly, we calculate the consumed energy and time of each cell (i.e., each task) in the table, and also at the same time calculate the total energy and execution time of this bit stream. However, if a random bit stream is generated which has some common cells with an existing string in the table, we only calculate the total energy of new string until the first common cell and then compare this new total energy with the existing total energy at this cell.

If the new total energy at this specific cell is less than the previous one, we keep the new sub-string and delete the old sub-string, and replace the total-energy and cell-energy of this cell with new amounts. We then update the energy and execution time of the remaining cells for the existing bit stream, based on the new values at this common cell. Otherwise, if the total energy of the existing bit stream is less than that of the new bit stream at the common cell, we will perform the same procedure while keeping the existing stream.

Every time a new stream is generated, we keep tracking the arrangement of the stream in the table. We terminate and accept a solution which has Hamming distance larger than a given threshold from an all 1 stream. The all 1 stream denotes the case where all components are executed locally. For example, the algorithm can terminate after K=20 iterations, or when 70% of the tasks have been offloaded. This heuristic termination criterion yields good results.

B. Algorithm of the Proposed Scheme The algorithm of our proposed scheme is shown in table II:

Table II. Algorithm of Proposed Method (DPH)

1.Initialize Energy and Time matrixes and set the time constraint
(Tconstratint) and Transmission Rate
2.for iteration = 1 to iteration_num
3.generate a random bit stream
4. check the first bit to specify the starting cell in the table
5. for i = 1 to N-1
6. Put each bit of the bit stream in the correct position in table
7. Calculate the self-Energy and time of each cell and the total energy and time.
8. if this specific cell in table is visited before compare the new Total Energy of this cell with the previous one
9. if the new Total Energy of the cell is less than the previous one
10. Replace the total energy and time of this Cell with the new calculated amounts.
11. Update the remaining amounts in the Remaining cells of the previous bit stream based on the new amount of this common cell.
12. Calculate the energy and time of the Remaining bits of the new bit stream.
13. Track the position of all bits in the table in a matrix
14. else
15. Keep the previous total energy and time in the cell.
16. Calculate the Energy and time of the remaining cells of the new stream based on the existing amount of this cell.
17. Track the position of all bits in the table in a matrix
18. End if
19. End if
20. End for
21.
22. if Number of bits in table = N & Etotal < Emin & Ttotal < Tconstraint & hamming distance criterion is met
23. return Etotal, Ttotal
24. end if
25. End for

IV. CONCLUSION

A mobile device must decide which computational tasks of a mobile application should be offloaded in order to minimize energy consumption while satisfying an execution time constraint. An efficient heuristic algorithm called DPH to solve this optimization problem is proposed, which uses dynamic programming combined with randomization.

It also uses a hamming distance as a termination criterion. Simulation results show that the proposed DPH algorithm can find nearly optimal solutions and it can be easily handle larger problems without losing computational efficiency. The DPH algorithm can be used dynamically, to adapt to the changes in the network transmission rate. The algorithm will tend to offload as many tasks as possible when the network performance is good, resulting in a rapid convergence to a near optimal solution with a very fast execution time.

REFERENCES

[1] Z. Li, C. Wang, and R. Xu, “Computation offloading to save energy on handheld devices: a partition scheme,” in Proc. International Conf. Compilers, Architecture, Synthesis Embedded Syst., pp. 238–246, 2001.

[2] P. Rong and M. Pedram, “Extending the lifetime of a network of batterypowered mobile devices by remote processing: a Markovian decisionbased approach” , Design Automation Conf., pp. 906–911, 2003.

[3] E. Cuervo, A. Balasubramanian, D.-k. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, “MAUI: Making smartphones last longer with code offload,” in Proc. ACM International Conference on Mobile Systems, Applications, and Services (MobiSys), pp. 49–62, 2010.

[4] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A Patti, “Clonecloud: Elastic execution between mobile device and cloud,” in Proc. ACM Conference on Computer Systems (EuroSys), pp. 301–314, 2011.

[5] S. Kosta, A. Aucinas, P. Hui, R. Mortier, and X. Zhang, “Thinkair: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading,” in Proc. IEEE International Conference

on Computer Communications (INFOCOM), pp. 945–953, 2012.

[6] Y. Liu, M. J. Lee, "An Effective Dynamic Programming Offloading Algorithm in Mobile Cloud Computing System", IEEE WCNC'14, pp.1868 – 1873, 2014.

[7] A. Toma, J. Chen, "Computation Offloading for Frame-Based Real- Time Tasks with Resource Reservation Servers", IEEE Euromicro Conference on Real-Time Systems, pp. 103-112, 2013.

[8] X. Gu, K. Nahrstedt, A. Messer, I. Greenberg, and D. Milojicic, "Adaptive offloading for pervasive computing," IEEE Pervasive Comput., vol. 3, no. 3, pp. 66–73, 2004.

[9]Available: http://darnok.org/programming/face-recognition/.

[10] A. Kammerdiner, P A. Krokhmal, P. M. Pardalos, "On the Hamming distance in combinatorial optimization problems on hypergraph matchings", Springer Optimization Letters, Vo. 4, pp. 609-617, 2010.

International Journal of Computer Applications Technology and Research
Volume 6–Issue 12, 498-500, 2017, ISSN:-2319–8656

Review of Efficient Techniques for Reducing Web Page Complexity

Anand M
Department of Information Science and Engineering,
GSSSIETW, Mysuru, India

***Abstract*-Users being frustrated due to high page load times. Because page load times are directly effects user satisfaction, providers would like to distinguish if and how the complexity of their Web sites affects the user experience. Although there is an extensive literature on measuring Web graphs, Website popularity, and the nature of Web traffic, there has been little work in understanding how difficult individual Web sites are, and how this complexity impacts the client's experience. We proposed a system that can identify a set of metrics to characterize the complexity of Web sites both at a content level and service level. Also the impact of complexity on the user performance and recommending what control measures should take to reduce the complexity.**

Keywords—Browsers, Websites, World Wide Web, Internet.

1. INTRODUCTION

Since many years, Web pages have become expressively more complex. At first web site used to host text and images, To rich media like Flash and Silverlight now the web pages contain several content type, ranging from video to script performed on the client's device. Additionally, a Web site today fetches content from servers hosted by its providers, and also from a range of third-party services such as advertising agencies, content distribution networks (CDNs), and analytics services. In combination, representation of a single Web page today involves fetching several objects with varying characteristics from multiple servers under different administrative domains.

In contrast, the poor effects of slow Web sites are well known. Users will discard or switch a Web Site due to performance issues. According to recent surveys, out of whole two thirds of users encounter slow Web Site. While abundance of anecdotal proof is that a key factor in slowing down Web Page is the increase in Web page complexity.Official studies on this topic have been limited. Most previous work on Web measurement concentrations on characterizing the Web graph [7], study the network footprint of Web traffic [2]-[5], also studying the rate of change of content on the Web [8]. Although these have contributed to a well understanding of Web usage, they do not examine the Web sites themselves.

In this paper we focuses on two broad questions,First,we count the complexity, called content level complexity of a Web page by means of a broad spectrum of metrics, and we characterize a Web page by the content in rendering like-the number of objects fetched, the sizes of these objects, and the types of content. Also we study the complexity of Web pages relating to the services they build upon. In addition to these we find number of bytes fetched and non-origin content accounts in place of a significant fraction of the number of objects

Our second and the main focus is on the time to download and render a Web page. We find that the total number of bytes fetched to render a Web site is the most dominant indicator of client-perceived load times than the number of objects fetched.

2. LITERATURE REVIEW

Michael Butkiewicz, Harsha V. Madhyastha, and VyasSekar [1] are focuse in these paper on finding the gap in understanding how complex individual Web sites are and how this complexity impacts on the usersperformance. Also characterize the Web site both at content level (like, number and size of images) and service level (like, number of servers/origins). It may happen that some categories are more complex than other such as 'News'. Out of hundred 60% of Web sites fetched content from minimum five non-origin sources, and these give more than 35% of the bytes downloaded. In addition, they examine which metrics are most suitable for predicting page render and load times and catch that the number of objects requested is the most important factor. With respect to variability in load times, however, they alsofind number of servers is the best indicator.

Y. Zhang, H. Zhu, and S. Greenwood [6] discuss about navigability. Navigability has become the axis of website designs. Existing mechanism haveproblem into two types. The major is to assess and measure a website's navigability in contrast to a set of principles. Another is to evaluate usage data of the Website. A metric methodology to Website navigability measurement is studies in this paper. Objectiveness and the probability of using automated tools to assess extensive websites are advantages of navigability metrics as far the existing valuation and analysis techniques.

Axiomatic assessment

Weyuker's axioms of software complexity have been frequently applied in place of a method to authorizing A logically the measurement of software complexity .In this section, they measure the metrics well-defined in the prior section compared to Weyuker's axioms of software complexity. Weyuker's axioms are established on a number of operators and relations on programs. According to the

features of websites these operators and relations must be modified.

M. Lee, R. R. Kompella, and S. Singh[3],in this paper disscuss on Cloud-based Web applications driven through new knowledge such as Asynchronous JavaScript and XML (Ajax) place an important load on network operators and creativities to effectively manage traffic. Problem happen is that there is no systematic technique to produce their workloads, notice their network performance today and possess track of the varying trends of these applications.they develop a tool, called AJAXTRACKER, that automatically impersonators a human interface with a cloud application and gathers associated network traces.

Ajax tracker

The main workings of AJAXTRACKER contain an event generator, a Web browser, a packet sniffer, and a traffic shaper. The event generator procedures the bulk of the tool.AJAXTRACKER is agnostic to the select of the Web browser and can work through any browser. The aim is to collect representative traces of a client session; packet sniffer captures the packets proceeding the client machine.

B. Krishnamurthy, C. E. Willis, and Y. Zhang [2],focuses on Content DistributionNetwork (CDNs). This is a tool to distribute contents just before and users. In content distribution origin server serves some or all the content of web pages. The technique like DNS resending and URL rewriting are balance the load among their servers. After the observation of results some CDNs provide better results as compare to other. In particular network the dramatic growth in the number of distinct server is gives best performance of one CDN company that can be improved between two testing periods. Either in average or worst case conditions the results shows that the case of DNS in the critical path of resource retrieval is not better than that of server choices related to client response.

2.5. Understanding Online Social Network Usage from a Network Perspective

F. Schneider, A. Feldmann, B. Krishnamurthy, and W.Willinger,focues in this paper study of Online Social Networks is discussed. Also, they understand which OSN feature inters and which one keep in consideration of poor users. Additionally the topics like friendship graph and sample crawls are studied on surveys. Extracting clickstreams as ofinactively observed network traffic these are the techniques using these they study how users are interact with OSN.

3. PROPOSED SYSTEM:

By using the “Characterizing Web Page Complexity and Its Impact” we are finding the complexity of web page. After the complexity has been calculated we can show the analytical reports in text as well as graphical format using graphs.

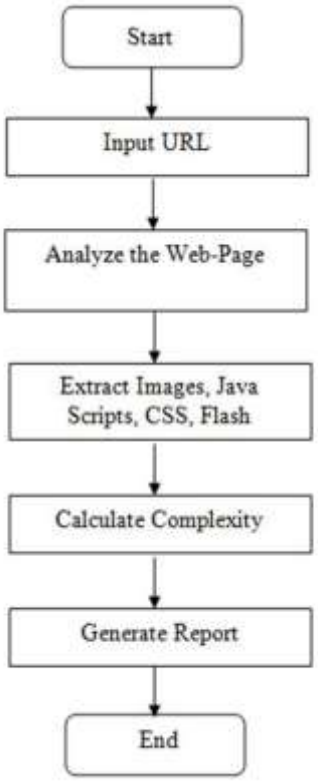


Figure1.Content Flow Architecture

We are starting with entering url after that the page is load. After the loading of page complexity of that web page can be determine using parameter such as content complexity and service complexity.

HTML5

HTML5 is a new standard for HTML which allows us to build rich and interactive web pages which bring HTML into the world of application development started in the year **2004**. HTML moves from simply describing the basics of a text based web for presenting audio, video and animations to enabling offline functionality, geo location and local storage in client side databases.

With the development of HTML5 it has wide range of applications in multimedia direction [3]. It can play audio and video and supports animations from the browser without the need of the proprietary technologies. The features of HTML5 would add up value for web designers and developers.

With the development of HTML5 it has wide range of applications in multimedia direction [3]. It can play audio and video and supports animations from the browser without the need of the proprietary technologies. The features of HTML5 would add up value for web designers and developers.

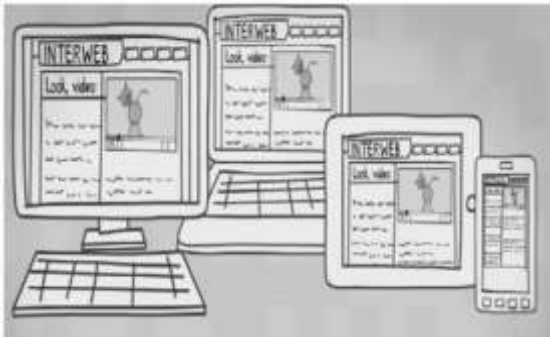


Fig 1. HTML5 cross-platform

HTML5 also support location based services open formats such as Scalable Vector Graphics (SVG), open XML file formats and high quality graphics.

The basic advantage for the developers and browsers is that they would be able to do more without the need of mastering or licensing multiple proprietary technologies that can develop rich web pages, enhanced forms and web based applications.

HTML5 FEATURES

HTML5 provides **new features** that include

1. Canvas – 2D/3D Graphics
2. Audio & Video
3. Location based Services
4. Working Offline
5. Web Workers

HTML5 supports cross platform, designed to display web pages on a PC, or a Tablet, a Smartphone, or a Smart TV (**Fig-1**). HTML5 is been a working draft and some browser designers and websites are already adopting HML5 elements.

1. Drag & Drop
2. New Input Types
3. New Elements
4. Form Elements

4. CONCLUSION:

We will develop an efficient system that will calculate the complexity of the webpage and identify the components of web pages which requires more time to load.

REFERENCES:

[1] Michael Butkiewicz, Harsha V. Madhyastha, and VyasSekar, “Characterizing Web Page Complexity and Its Impact”, VOL. 22, NO. 3, JUNE 2014

[2] B. Krishnamurthy, C. E. Willis, and Y. Zhang, “On the use and performance of content distribution networks” , in Proc. IMW, 2001, pp. 169–182.

[3] M. Lee, R. R. Kompella, and S. Singh, “Active measurement system for high-fidelity characterization of modern cloud applications”, in Proc. USENIX Conf. Web Appl., 2010, p. 2.

[4] F. Schneider, A. Feldmann, B. Krishnamurthy, and W.Willinger, “Understanding online social network usage from a network perspective”, in Proc. IMC, 2009, pp. 35–48.

[5] F. Schneider, S. Agarwal, T. Alpcan, and A. FeldmanninProc. PAM ,“The new Web:Characterizing AJAX traffic”, 2008, pp. 31–40

[6] Y. Zhang, H. Zhu, and S. Greenwood, “Website complexity metrics for measuring navigability,” in Proc. Int. Conf. Quality Softw., 2004, pp.172–179

[7] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener, Comput. Netw., “Graph structure in the Web,” vol. 33, no. 1, pp. 309–320, Jun. 2000.

[8] D. Fetterly, M. Manasse, M. Najork, and J. Wiener, “A large-scale study of the evolution of Web pages,” in Proc. WWW, 2003, pp.669–678.

International Journal of Computer Applications Technology and Research
Volume 6–Issue 12, 501-510, 2017, ISSN:-2319–8656

A Cortical Learning Movement Classification Algorithm for Video Surveillance

Abdullah Alshaikh
Staffordshire University
School of Computing and
Digital Technologies
Stoke-on-Trent, UK

Mohamed Sedky
Staffordshire University
School of Computing and
Digital Technologies
Stoke-on-Trent, UK

Abstract: Classifying the movements of objects detected from a video feed is a key module to achieve a cognitive surveillance system. Machine learning techniques have been heavily proposed to solve the problem of movement classification. However, they still suffer from various limitations such as their limited ability to learn from streamed data. Recently, Hierarchical Temporal Memory (HTM) theory has introduced a new computational learning model, Cortical Learning Algorithm (CLA), inspired from the neocortex, which offers a better understanding of how our brains process temporal information. This paper proposes a novel biologically-inspired movement classification algorithm based on the HTM theory for video surveillance applications. The proposed algorithm has been tested using twenty-three videos, from VIRAT dataset, and an average accuracy of 85% was achieved.

Keywords: hierarchical temporal memory, cortical learning algorithms, movement classification, video forensic, post incident analysis

1. INTRODUCTION

Movement classification algorithms aim at learning motion patterns of objects of interest in a surveillance scenario to classify a new movement. They also attempt to understand the trajectories of tracked objects and the interactions between them. The application domain where these algorithms are engaged is a cognitive surveillance system 1].

Several movement classification methods have been proposed in the literature, which has several merits and demerits, but the advances in these methods have continued and are recently gaining importance and attention of many researchers due to the need for flexible, adaptable ways of solving movement classification problems [2].

Numerous computation techniques have been introduced to enhance computation beyond the physical limits of computers for solving complex problems. One such approach is called biologically inspired computing, also known as a bio-Inspired approach. 'Learning from experience' is a basic task of the human brain which is not yet fulfilled satisfactorily by computers. Moreover, recently cope with this issue, several researchers have been involved in bio-inspired approaches, where a learning method is proposed based on a model derived from neurophysiological observations of the generation of the sense of self which is connected to the memorisation of the interaction with external entities. Therefore, bio-inspired algorithms are based on the structure and functioning of complex natural systems and tend to solve problems in an adaptable and distributed fashion.

New bio-inspired machine learning techniques have been proposed in the attempt of mimicking the function of a human brain. Hierarchical Temporal Memory (HTM) theory has proposed new computational learning models, Cortical Learning Algorithms (CLA), inspired from the neocortex, which offer a better understanding of how our brains function. HTM gives an adaptable and naturally precise system for settling expectation, grouping, and oddity location issues for a wide scope of information sorts [3, **Error! Reference source not found.**].

This paper proposes a novel bio-inspired movement classification algorithm based on the CLA. The proposed algorithm can be used to automate video analytics and video forensic systems.

The remaining of the paper is structured as follows: Section 2 presents a review of related works in movement classification, it includes literature that studies the CLA. The proposed movement classification technique is presented in section 3. The used dataset and the evaluation criteria are presented in section 4. The results are analysed in section 5, and discussed. The paper concludes with section 7 by drawing insights from the proposed techniques, experimentation and results.

2. PREVIOUS WORK

The use of video analytic technologies has gained wide attention in the research community and the global security around the world [5]. The purpose of intelligent visual surveillance in most cases is to learn, detect and recognise interesting events that seem to constitute challenges to the

community or area of the target [6]. These challenges posed by defining and classifying events as unusual behaviour [7], abnormal behaviour [8], anomaly [9] or irregular behaviour [10]. Activity recognition techniques are reviewed, in [1].

Biologically inspired algorithms or bio-inspired algorithms for classification are a class of algorithms that imitate specific phenomena from nature. Bio-inspired algorithms are usually bottom-up, decentralized approaches which specify a basic set of conditions and rules that attempt to solve a complex problem by iteratively applying them. Such algorithms aim to be adaptive, reactive and distributed fashion [14]

2.1 Cortical Learning Algorithms

Cortical Learning Algorithms (CLAs) comprises an effort by Numenta Incorporation [25] to design a model that can perceptually and computationally analyse neocortex learning in the brain. The cortical learning algorithms are utilized as a part of the second implementation of a designed framework for perceptual learning called Hierarchical Temporal Memory (HTM). The algorithm, CLA, functions on a set of data structure, and the two of them together accomplish some level of spatial and temporal pattern recognition. The data structure utilised is a gathering of segments of cells, called a locale. A cell in a section is a neuron-like substance, which makes associations with different cells, and totals their action to decide its state of initiation.

It is biologically proven that neocortex is the seat of intelligent thought in the human or mammalian brain. Intelligent properties such as vision, movement, hearing, touching, etc. are all performed by this intelligent seat, this cognitive tasks that are primarily performed by the neocortex of humans are challenging to design in real life scenarios.

2.1.1 Types of CLA Components

The CLA consists of four main components: Encoder, Spatial Pooler, Temporal Memory and a classifier

Encoder: The initial step of utilising an HTM framework is to change from an information source into a Sparse Distributive Representations (SDRs) using an encoder. The encoder changes over the local configuration of the information into an SDR that can be bolstered into an HTM framework. The encoder is in charge of figuring out which bits ought to be ones, and which ought to be zeros, for a given information esteem in such a route as to catch the essential semantic qualities of the information. Comparative information qualities ought to deliver overlapping SDRs [**Error! Reference source not found.**17]. HTM frameworks require information contribution to the type of SDRs [**Error! Reference source not found.**]. An SDR comprises of a vast exhibit of bits of which most are zeros. The encoder aims to generate a code where every piece conveys some semantic meaning so if two SDRs have more than a couple overlapping one-bits, then those two SDRs have comparable implications.

Spatial pooling: The open field of every section is a settled number of information sources that are arbitrarily chosen from a much bigger number of hub data sources. Considering the info design, a few segments will get more dynamic information values. Spatial pooling chooses a consistent number of the most dynamic sections and inactivates (represses) different segments in the region of the dynamic ones. Comparable information designs tend to actuate a steady arrangement of sections.

Temporal Memory: Temporal memory has been a dynamic region of research for HTMs. The significance of temporal memory and the general objectives of temporal pooling have been to a great extent predictable [19]. Be that as it may, the expression "temporal memory" has been utilised for various diverse executions and looking through the code, and past documentation can be to some degree confounding. The first CLA Whitepaper utilised the term temporal pooler to depict a specific usage. This usage was unpredictably tied in with succession memory. Like this the succession memory and transient pooling were both alluded to as "temporal pooling", and the two capacities were perplexed [19].

Classifier: HTM-CLA plans to learn and speak to structures and groupings in light of memory predictions. In any case, the classifier used to interpret the arrangement yield from HTM-CLA are a long way from palatable. Classifiers utilised as a part of the NuPIC structure are KNN, CLA and SDR Classifiers [20]. Two new classifiers are also proposed by [20] given various similitude assessment strategies. The principal technique is H-DS Classifier given Dot Similarity and the second strategy is H-MSC Classifier given Mean-Shift Clustering, in an attempt to make the classifiers in HTM-CLA more productive and powerful.

2.1.3 The Choice of CLA

CLA being an online learning algorithm and needs no pre-processing and requires less training time. For example, CLA has been applied to solve the problem of classifying Electrocardiogram (ECG) samples into sick and healthy groups discriminating subsequence eliminated in the signal after supervision which could otherwise be done by the human supervisor [29].

This paper proposes a bio-inspired Movement Classification technique that tends to achieve an efficient and effective performance.

3. PROPOSED MOVEMENT CLASSIFICATION TECHNIQUE

The proposed bio-inspired movement classification is based on the CLA that learns to predict a sequence of movements. A slightly erroneous copy of the learned sequences will be presented to the algorithm, which will recover quickly after any unexpected or suspicious movement patterns.

However, going to predict the rest of the sequence, this would be a desirable property since real-world data is likely to be noisy and dynamic. The proposed Cortical learning movement classification algorithm presents a unique and novel way of approaching this problem.

3.2 Movement Classification Datasets

Most post-incident analysis cases target outdoor scenarios. Not all publicly available movement classification and action recognition datasets represent realistic real-world surveillance scenes and scenarios as they contain short clips that are not representative of expected actions in these scenarios. Some of them provide limited annotations which comprise event examples and trajectories for moving objects, and hence lack a solid basis for evaluations in large-scale.

VIRAT video dataset is a large-scale dataset that facilitates the assessing of movement classification algorithms. The dataset used for this study was designed to be natural, realistic, and challenging for video surveillance domains stipulated to its background clutter, resolution, human event/activity categories and diversity in scenes than existing action recognition datasets [28].

According to [28] the dataset distinguishing characteristics are as the following:

- **Realism and natural scenes:** VIRAT's data is collected in natural scenes by showing people in standard contexts performing normal actions, with cluttered backgrounds in an uncontrolled environment.
- **Diversity:** VIRAT's data is collected from multiple sites through a variety of camera resolutions and viewpoints, while many different people perform actions.
- **Quantity:** Various types of human-vehicle and human actions interaction are included with a large number of examples (>30) per action class.
- **A wide range of frame rates and resolution:** Many applications operate across a wide range of temporal and spatial resolutions such as video surveillance. Therefore, the dataset is designed purposely to capture the ranges, (with 2–30Hz) frame rates and 10–200 pixels in person-height.

4. Dataset and Evaluation Criteria

VIRAT dataset includes a total of eleven scenes that were recorded in the videos captured by

stationing high definition cameras. Due to the wind, the videos reordered might experience a little clutched and encoded in H.264 as highlighted from VIRAT Dataset. Each scene contains many video clips, and each clip has zero or many instances. The file name format is unique which makes it easier for the identification of videos that are from the same scene using the last four digits that indicate collection group ID and scene ID.

4.1 Annotation Standard

There is a total of twelve different types of events which are either fully annotated or partially annotated. The event is represented as the set of activities objects are involved within a time interval, e.g. "PERSON loading an OBJECT into a VEHICLE" and "PERSON unloading an OBJECT from a VEHICLE". Objects are annotated as long as they are within the vicinity of the camera and stop recording a few seconds after the object is out of the vicinity of the camera, all this and much more are considered in terms of analysis and evaluation purposes. MATLAB software is used for this evaluation.

VIRAT dataset includes two sets of annotation files that describe (a) the objects and (b) the events depicted in the videos. Samples of the event annotation files and the object annotation files are shown in Table 4-1 and Table 4-2 these annotation files were generated manually and represent the ground truth used for evaluation. The training includes 66 videos representing three scenes.

The events included in VIRAT training dataset are:

- *unknown=0,*
- *loading=1,*
- *unloading=2,*
- *opening_trunk=3,*
- *closing_trunk=4,*
- *getting_into_vehicle=5,*
- *getting_out_of_vehicle = 6.*

Table 4-1 Sample of VIRAT's object annotation file

Object ID	Duration of object	Frame number	bbox X_lt	bbox Y_lt	bbox Width	bbox Height	Object Type
1	385	3495	157	659	76	132	1
1	385	3496	162	658	76	132	1
. .	. .	. .	. .	. .	. .	. .	. .
1	385	3838	747	498	73	97	1
1	385	3839	747	498	73	97	1
3	4732	0	613	469	254	189	2
3	4732	1	612	468	255	190	2
. .	. .	. .	. .	. .	. .	. .	. .

Object Type: type of object (Unknown=0, person=1, car=2, other vehicle=3, other object=4, bike=5)

4.2 Combining the two files

A Matlab script has been developed to generate a file that combines information from VIRAT object annotation files with corresponding information from VIRAT events annotation files for each video file. Table 4-3 shows the combination of the events and objects annotation file obtained from the sample VIRAT dataset.

Table 4-2 Sample of VIRAT training dataset object annotation file

Reset	Event-ID	Frame No.	Event Type	Object Type	Object ID	bbox X_lt	bbox Y_lt	bbox	bbox
1	**0**	**0**	**6**	**2**	**2**	**648**	**497**	**154**	**66**
0	**0**	**1**	**6**	**1**	**1**	**720**	**490**	**26**	**22**
0	**0**	**1**	**6**	**2**	**2**	**648**	**497**	**154**	**66**
..	..	..	..	..	..	..	..	..	..
1	**1**	**0**	**3**	**2**	**1**	**457**	**432**	**93**	**58**
0	**1**	**0**	**3**	**1**	**2**	**533**	**479**	**21**	**48**
0	**1**	**0**	**3**	**2**	**3**	**205**	**371**	**71**	**44**

4.3 Performance Evaluation

Scene-independent and scene-adapted learning recognitions are the two evaluation modes that are used for testing datasets. Scene-independent has a trained event detector on the scene which is not included in the test, while scene-adapted recognition applied to the clips that may be used for training processes, but the test clips are not used during the process.

This evaluation is based on the documents from VIRAT dataset release 2.0 [27] [28]. This document from the VIRAT dataset website has the following contents that are described below

4.4 EXPERIMENTAL SET-UP

Table 4-3 Sample of the combined generated data

E-ID	E-Ty	E-Lg	E-S-Fr	E-E-Fr	bbox X_lt	bbox Y_lt	bbox Width	bbox Height	NOO
1	5	172	3670	3841	670	454	267	228	2
2	5	217	10413	10629	985	406	209	204	2
3	2	66	10068	10133	891	357	202	128	3
4	6	131	9614	9744	983	399	226	211	2
5	6	112	15812	15923	1220	378	241	126	2
6	5	151	17522	17672	1253	380	198	126	2
2	5	217	10413	10629	985	406	209	204	2

The test egins by isolating the training part of VIRAT vdeo dataset into two sections. The initial segment is utilised for training purposes and the second part is utilised for testing, 60% of the data has been utilised for training and the remaining has been utilised for testing. Each try begins by moving one event or two events, from the training dataset to the testing dataset. The record name demonstrates the shrouded event which has been moved to the testing dataset and is not shown to the algorithm in the training phase e.g. Event0, Event1, Event2 .. Event6. The point is to conceal those occasions in the preparation and to present them in the testing to discover how the proposed algorithm can identify a new event as an anomaly.

4.5 Data Preparation

The data preparation starts by moving one event or two events to the end of the file and the purpose of that to hide those events during the training phase and present it in the testing phase to find out how the system has learnt and understood from previous events.

The results of the CLA anomaly detection algorithm is represented by an anomaly score for each field. This score varies between Zero and One. Where values close to Zero represent movements closer to normal ones and values closer to one represent movements that are abnormal.
First, the evaluation starts from the first test field until a first record that represents an event, which has been hidden in training, appears. The accuracy is calculated by comparing the resulted anomaly score with a threshold. If the anomaly score is less than the threshold the movement is considered normal.

The second step starts when the first record of a hidden event appears. In this case if the resulted anomaly score is greater than the threshold, the result is considered correct. This process has been repeated for threshold values between 0.1 and 0.9 with a step of 0.1 to find the maximum accuracy and hence to identify the optimum threshold.

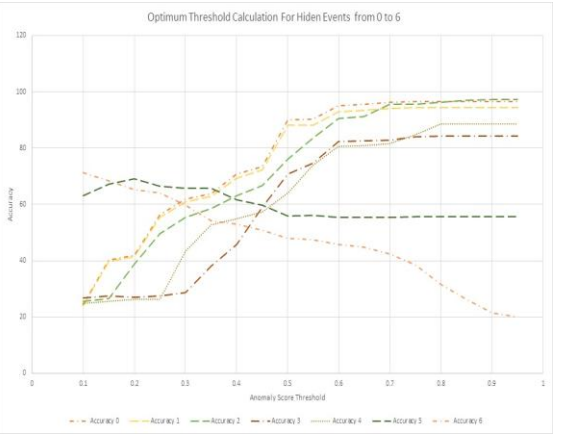
The mathematical equation below defines the calculated accuracy.

$$Accuracy = \frac{Total\ number\ of\ correct\ detection}{The\ total\ numbers\ of\ tested\ events}$$

5. DATASETS ANALYSIS RESULT (CLA ANOMALY ALGORITHM)

Figure 1 highlights the various accuracy trends of the proposed algorithm. The figure shows the change of those anomalies are with different threshold scores.

Figure 1: Optimum Threshold Calculation for Hidden Events from 0 to 6



predicted events where the blue bars refers to the ground truth or the VIRAT dataset, and the orange bars to the predicted ones.
The second machine learning is Decision Tree Learning [25], which is one of the predictive modelling approaches used in statistics, data mining and machine learning. Tree models where the target variable can take a discrete set of values are called classification trees. In these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels.

6. Conclusion

Artificial Intelligence (AI) and Neural Networks (NN) have been widely used to solve the movement classification problem. However, they still have various limitations for instance, their limited scope of operations. In the attempt of mimicking the function of a human brain, learning models inspired from the neocortex has been proposed which provide better understating of how our brains function. Recently, new bio-inspired learning techniques have been proposed and their results have shown evidence of superior performance over traditional techniques. The CLA processes streams of information, classify them, learning to spot the differences, and using time-based patterns in order to predict. In humans, these capabilities are mainly performed

by the neocortex. Hierarchical Temporal Memory (HTM) is a technology modelled on how the neocortex performs these functions. HTM provides the promise of building machines that approach or exceed the human level performance for many cognitive tasks.

The proposed Bio-Inspired movement classification technique is based on HTM and is biologically used to solve many problems looking at the set of requirement that bio-inspired movement classification technique uses.

The conclusions, from this study, which is drawn are stated below: -

1. The neocortex inspired learning techniques was suitable for correctly learning and predicting a sequence of movement and can then be presented with a slightly erroneous copy of the sequence, which will cover quickly after any unexpected or suspicious movement patterns.

2. As it is also going to predict the rest of the sequence, this would be a desirable property since real world data is likely to be noisy and dynamic in nature.

This study has given indications that neocotex inspired learning techniques are applicable for activities in movement classification aspect of the analysis of video forensic evidence.

7. REFERENCES

1. Vishwakarma, S. and Agrawal, A., 2013. A survey on activity recognition and behavior understanding in video surveillance. The Visual Computer, 29(10), pp.983-1009.

2. Forbes, N., 2000. Biologically inspired computing. Computing in Science & Engineering, 2(6), pp.83-87.

3. Hawkins, J., & Ahmad, S. (2015). Why Neurons Have Thousands of Synapses, A Theory of Sequence Memory in Neocortex. arXiv preprint arXiv:1511.00083.

4. Ahmad, S., & Hawkins, J. (2016). How do neurons operate on sparse distributed representations? A mathematical theory of sparsity, neurons and active dendrites. arXiv preprint arXiv:1601.00720.

5. Popoola, P. and Wang, J. (2012) 'Video-Based Abnormal Human Behaviour Recognition-A Review' IEEE Transaction on Systems, MAN, and Cybernetics-Part C: Applications and Review, vol. 42, no. 6, Nov., 2012.

6. Lavee, G. Khan, L. and Thuraisingham, B (2007) 'A framework for a video analysis tool for suspicious event detection' Multimedia Tools Appl., vol. 35, pp. 109–123, 2007.

7. Hara, K. Omori, T. and Ueno, R (2002) 'Detection of unusual human behavior in intelligent house' in Proc. 2002 12th IEEE Workshop Neural Netw. Signal Process., 2002, pp. 697–706.

8. Lee, C. K. Ho, M. F. Wen, W. S. and Huang, C.L (2006) 'Abnormal event detection in video using N cut clustering' in Proc. Int. Conf. Intell. Inf. Hiding Multimedia Signal Process., 2006, pp. 407–410.

9. Feng P. and Weinong, W (2006) 'Anomaly detection based on the regularity of normal behaviors' in Proc. 1st Int. Symp. Syst. Control Aerosp. Astronautics, Jan.19–21, 2006, pp. 1041–1046.

10. Zhang Y. And Liu, Z (2007) 'Irregular behavior recognition based on trading track' in Proc. Int. Conf. Wavelet Anal. Pattern Recog., 2007, pp. 1322–1326.

11. Kobayashi, M., Okabe, T. and Sato, Y. 2010. Detecting forgery from static-scene video based on inconsistency in noise level functions, Information Forensics and Security, IEEE Transactions on 5 (4) (2010) 883{892.

12. Jing Zhang , Yuting Su , Mingyu Zhang, 2009. Exposing digital video forgery by

ghost shadow artifact, Proceedings of the First ACM workshop on Multimedia in forensics, October 23-23, 2009, Beijing, China

13. Wang, W. and Farid, H. 2009. Exposing digital forgeries in the video by detecting double quantization, in: Proceedings of the 11th ACM workshop on Multimedia and security, ACM, 2009, pp. 39-48.

14. Ding, S., Li, H, Su, C., and Yu, J. 'Evolutionary artificial neural networks: a review' Artif Intel Rev, 2011, vol 39, pp 251-260.

15. Costello, C.J. and Wang, I., 2005, December. Surveillance camera coordination through distributed scheduling. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on* (pp. 1485-1490). IEEE.

16. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T. and Serre, T., 2011, November. HMDB: a large video database for human motion recognition. In*Computer Vision (ICCV), 2011 IEEE International Conference on* (pp. 2556-2563). IEEE.

17. Purdy, S., 2016. Encoding data for HTM systems. arXiv preprint arXiv:1602.05925.

18. Bosch, A., Zisserman, A. and Muoz, X. (2008) Scene classification using a hybrid generative/discriminative approach. IEEE Trans. Pattern Analysis and Machine Intell., 30(04):712–727, 2008.

19. Melis, W.J., Chizuwa, S. and Kameyama, M., 2009, May. Evaluation of the hierarchical temporal memory as a soft computing platform and its VLSI architecture. In Multiple-Valued Logic, 2009. ISMVL'09. 39th International Symposium on (pp. 233-238). IEEE.

20. Zhituo, X., Hao, R. and Hao, W., 2012, October. A Content-Based Image Retrieval System Using Multiple Hierarchical Temporal Memory Classifiers. In Computational Intelligence and Design (ISCID), 2012 Fifth International Symposium on (Vol. 2, pp. 438-441). IEEE.

21. Balasubramaniam, J., Krishnaa, C. G., & Zhu, F. (2015). Enhancement of Classifiers in HTM-CLA Using Similarity Evaluation Methods. Procedia Computer Science, 60, 1516-1523.

22. Ermoliev, Y., 1983. Stochastic quasigradient methods and their application to system optimization. Stochastics: An International Journal of Probability and Stochastic Processes, 9(1-2), pp.1-36.

23. Bottou, L., 2010. Large-scale machine learning with stochastic gradient descent. In Proceedings of COMPSTAT'2010 (pp. 177-186). Physica-Verlag HD.

24. Schapire, R.E., 2003. The boosting approach to machine learning: An overview. In Nonlinear estimation and classification (pp. 149-171). Springer New York.

25. Rodriguez, M., Orrite, C., Medrano, C. and Makris, D., 2016. A time flexible kernel framework for video-based activity recognition. Image and Vision Computing, 48, pp.26-36.

26. George, D. and Hawkins, J., 2009. Towards a mathematical theory of cortical micro-circuits. PLoS computational biology, 5(10), p.e1000532.

27. Moon, J., Kwon, Y., and Kang, K., 2015. ActionNet-VE Dataset: A Dataset for Describing Visual Events by Extending VIRAT Ground 2.0. *2015 8th International Conference on Signal Processing, Image Processing and Pattern Recognition (SIP)*

28. Oh, S., Hoogs, A., Perera, A., Cuntoor, N., Chen, C., Lee, J. T., Mukherjee, S., Aggarwal, J. K., Lee, H., Davis, L., Swears, E., Wang, X., Ji, Q., Reddy, K., Shah, M., Vondrick, C., Pirsiavash, H., Ramanan, D., Yuen, J., Torralba, A., Song, B., Fong, A., Roy-Chowdhury, A., and Desai, M. 2011. A Large-scale Benchmark Dataset for Event Recognition in Surveillance Video. *In Proceedings of IEEE Comptuer Vision and Pattern Recognition (CVPR), 2011.*

29. Akrami, A., Akrami, A., Solhjoo, S. & Nasrabad, A. (2005). EEG-Based Mental Task Classification: Linear and Nonlinear Classification of Movement Imagery. *In Engineering in Medicine and Biology 27th Annual Conference. Shanghai, China, pp. 4626–4629.*

International Journal of Computer Applications Technology and Research
Volume 6–Issue 12, 511-515, 2017, ISSN:-2319–8656

A survey on speech signal synthesis system

Soumya Dath G
Assistant Professor, Dept. Of ISE
GSSSIETW, Mysuru, Karnataka, India

Abstract: The primary objective of this paper is to provide an overview of existing methods Text-To-Speech synthesis techniques. Text to speech synthesis can be broadly categorized into three categories, formant Based, Concatenative based and Articulatory. Formant based speech synthesis relies on different techniques such as cascade, parallel, klatt and PARCAS Model etc. Concatenative speech synthesis can be broadly categorized into three categories, Diphones Based, Corpus based and Hybrid whereas Articulatory synthesis involves Vocal Tract Models, Acoustic Models, Glottis Models , Noise Source Models . In this paper, all text to speech synthesis methods are explained with their pros and cones.

Keywords: Text to speech synthesis, Formant speech synthesis, Concatenative speech synthesis, Articulatory speech synthesis

1. INTRODUCTION

Text-to-speech (TTS) synthesis ultimate goal is to create natural sounding speech from arbitrary text. Moreover, the current trend in TTS research calls for systems that enable production of speech in different speaking styles with different speaker characteristics and even emotions. Speech synthesis generally refers to the artificial generation of human voice – either in the form of speech or in other forms such as a song. The computer system used for speech synthesis is known as a speech synthesizer. There are several types of speech synthesizers (both hardware based and software based) with different underlying technologies. For example, a TTS (Text to Speech) system converts normal language text into human speech, while there are other systems that can convert phonetic transcriptions into speech. The goal of a text-to-speech system is to automatically produce speech output from new, arbitrary sentences. The text-to-speech synthesis procedure consists of two main phrases. The first is text analysis, in which the input text is transcribed into a phonetic or some other appropriate representation, and the second is the actual generation of speech waveforms, in which the acoustic output is produced from the information obtained from the first phase [2]. A simplified version of the synthesis procedure is presented in figure 1.

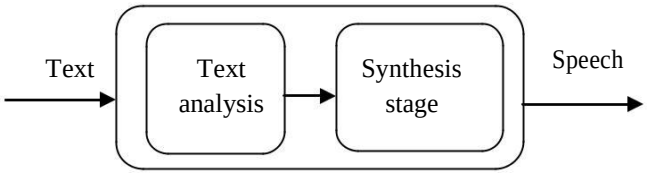


Figure 1: Phases of Text -to -speech system

The quality of a speech synthesizer is measured based on two primary factors – its similarity to normal human speech (naturalness) and its intelligibility (ease of understanding by the listener). Ideally, a speech synthesizer should be both natural and intelligible, and speech synthesis systems always attempt to maximize both characteristics [3].

A typical text to speech system has two parts – a front end and a back end. The front end is responsible for text normalization (the pre-processing part) and text to phoneme conversion. Text normalization or tokenization is the phase where numbers and abbreviations in the raw text are converted into written words.

Text to phoneme conversion or grapheme-to-phoneme conversion is the process of assigning phonetic transcriptions to each word and dividing them into prosodic units such as phrases, clauses, and sentences. The output of the front-end system is the symbolic linguistic representation of the text. It is composed of the phonetic transcriptions along with the prosody information. This output is then passed on to the back-end system or the synthesizer, which converts it into sound [1].

This paper is organized as follows. This section gives an introduction about text to speech synthesis. In section II, a review about various methods for text to speech synthesis is explained in detail. The conclusion is given in section III.

2. METHODS OF TEXT TO SPEECH SYNTHESIS

Various methods of text to speech synthesis are explained below.

2.1 Formant Synthesis

Formant synthesis is based on the source-filter-model of speech. There are two basic structures: parallel and cascade, but for better performance some kind of combination of these is usually used. Formant synthesis also provides infinite number of sounds which makes it more flexible than concatenation methods. In this approach, at least three formants are generally required to produce intelligible speech and to produce high quality speech up to five formants are used. Each formant is modelled with a two-pole resonator which enables both the formant frequency (pole-pair frequency) and its bandwidth to be specified. The input parameters may be the open quotient that means the ratio of the open-glottis time to the total period duration: Voicing fundamental frequency (F0), Voiced excitation open quotient (OQ) , Degree of voicing in excitation (VO), Formant frequencies and amplitudes (F1...F3 and A1...A3), Frequency of an additional low-frequency resonator (FN) and Intensity of low and high-frequency region (ALF, AHF)[1].

2.1.1 A Cascade Formant Synthesizer:

It consists of band-pass resonators connected in series and the output of each formant resonator is applied to the input of the next one. The cascade structure needs only formant frequencies as control information. The main advantage of the

cascade structure is that the relative formant amplitudes for vowels do not need individual controls.

The cascade structure is better for non-nasal voiced sounds because it needs less control information than parallel structure. Moreover, it is then simple to implement. However, with cascade model the generation of fricatives and plosive bursts is a problem [1].

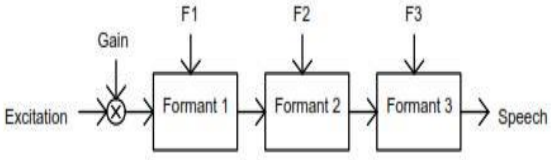


Fig. 2 Basic structure of cascade formant synthesizer [1]

2.1.2 A Parallel Formant Synthesizer:

It consists of resonators connected in parallel. Sometimes extra resonators for nasals are used. In this method, the excitation signal is applied to all formants simultaneously and their outputs are summed. Adjacent outputs of formant resonators must be summed in opposite phase to avoid unwanted zeros or anti resonances in the frequency response. The parallel structure enables controlling of bandwidth and gains for each formant individually and thus needs more control information [1].

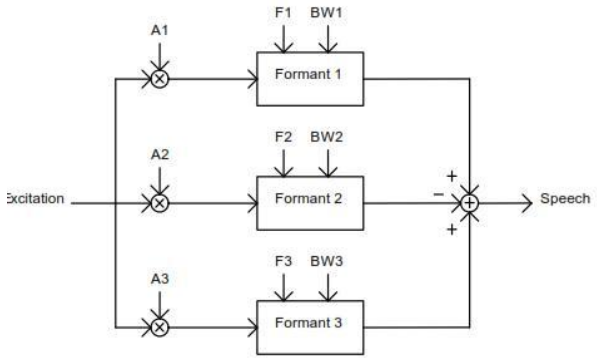


Fig. 3 Basic structure of a parallel formant synthesizer [1]

The parallel structure has been found to be better for nasals, fricatives, and stop consonants, but some vowels cannot be modelled with parallel formant synthesizer as well as with the cascade one, this is the main disadvantage of it.

2.1.3 Klatt Formant Synthesizer:

To see the good results with only either cascade or parallel formant synthesiser method is difficult. So to improve results, combination of these basic models is used. In 1980 Dennis Klatt proposed a more complex formant synthesizer which incorporated both the cascade and parallel synthesizers with additional resonances and anti-resonances for nasalized sounds, sixth formant for high frequency noise, a bypass path to give a flat transfer function, and a radiation characteristics. This system used quite complex excitation model which was controlled by 39 parameters updated every 5 ms. The quality of Klatt Formant Synthesizer was very promising and this model has been incorporated into several present TTS systems, such as MITalk, DECtalk, Prose-2000, and Klattalk[1].

2.1.4 PARCAS (Parallel-Cascade) model:

In the model, the transfer function of the uniform vocal tract is modelled with two partial transfer functions, each including every second formant of the transfer function. Coefficients k1, k2, and k3 are constant and chosen to balance the formant amplitudes in the neutral vowel to keep the gains of parallel branches constant for all sounds[1].

The PARCAS model uses a total of 16 control parameters: F0 and A0 - fundamental frequency and amplitude of voiced component, Fn and Qn - formant frequencies and Q-values (formant frequency / bandwidth), VL and VH - voiced component amplitude, low and high, FL and FH - unvoiced component amplitude, low and high, QN - Q-value of the nasal formant at 250 Hz[1].

The used excitation signal in formant synthesis consists of some kind of voiced source or white noise. The correct and carefully selected excitation is important especially when good controlling of speech characteristics is wanted. The formant filters represent only the resonances of the vocal tract, so additional provision is needed for the effects of the shape of the glottal waveform and the radiation characteristics of the mouth. Usually the glottal waveform is approximated simply with -12dB/octave filter and radiation characteristics with simple +6dB/octave filter[1].

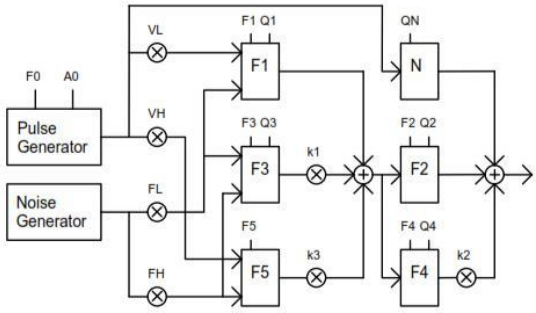


Fig. 4 PARCAS model [1]

2.2 Concatenative Speech Synthesis

Concatenative speech synthesis method involves production of artificial speech by concatenating pre recorded units of speech by phonemes, diphones, syllabus, words or sentences. This method involve the selection of appropriate units from the speech database and algorithms that join selected units and perform some signal processing to smoothen the concatenation the boundary. Here, speech is produced by selecting and concatenating appropriate speech unit from speech database where speech data base consists of speech units of different sizes such as phones, di-phones, syllables, words/sentences. Concatenative speech synthesis can be done by three different methods [2].

2.2.1 Di phone based speech sythesis:

Diphone is used as a basic speech unit for this synthesis method where diphone is two connected half phones starting in the middle of first phone and ending in the middle of second phone.

As only single instances of all speech units are available in the speech database, to obain good quality of synthesiesd speech with the procody, various signal processing methods are applied. PSOLA, TD – PSOLA, LP – PSOLA, ESNOLA, FD-

PSOLA are various signal processing methods are used for obtaining good quality synthesised speech.

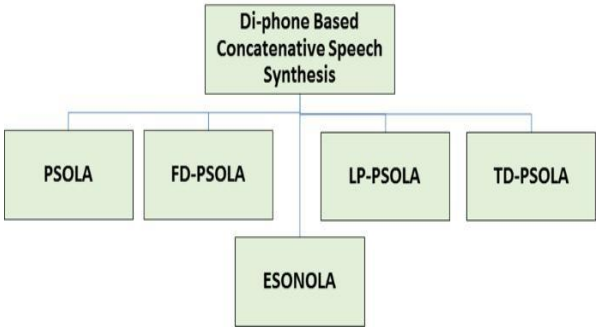


Fig 5. Various Signal Processing Techniques Used in Di-phone Based Speech Synthesis

2.2.1.1 Pitch synchronous Overlap Add(PSOLA):

This method is analysis synthesis method, that involves decomposition of speech signal into number of pitch short synchronised waveforms.To obtain multiple synthesis, short term signal the pitch synchronous short term waveform can be altered time or spectral domain and final synbthesised speech is produced by overlapped addition of this short term signal.It preserves the spectral envelope when pitch shifting is used. This is the main advantage of this method. In addition it does not lose any information of signal, since it work directly on the signal.

2.2.1.2 Frequency Domain Pitch Synchronous Overlap Add (FD-PSOLA):

This method is used for modifying the spectral envelope and for compute spectral envelope it uses linear prediction and pitch is modified by linear interpolation on the spectral envelope.

In this method, all operations are performed in frequency domain and main advantage of this system is easy implementation. FDPSOLA leads discontinuities at the concatenation boundary. Moreover, it is computationally intensive and has high memory requirements for storage which are the main disadvantages of systems.

2.2.1.3 Linear Prediction Pitch synchronous overlap Add (LP-PSOLA):

By manipulating the LP residual, modification of pitch and duration of speech is achieved. This method is suitable for pitch scale modification as it provides independent control over spectral envelop for synthesis.

Due to overlap and adding of the windowed residual segments, it producing phase mismatch and audible distortion, this is main disadvantage of this method.

2.2.1.4 Time Domain Pitch Synchronized Overlap Add (TD-PSOLA):

This method is used for the modification of the prosody of speech waveforms, as it facilitates the high quality pitch and time scale modification. This method is computationally efficient method but it has a drawback of large speech database requirement. Moreover, the quality of synthesized speech is affected by the detection of epochs in the speech signal which is very difficult to achieve in in real time applications.

2.2.1.5 Epoch Synchronous Non Overlap Add (ESNOLA):

In this method, the synthesized speech is generated by concatenating basic speech segments at the epoch positions of the voiced speech, where epochs represent quasi periodic sounds. Pitch and duration modifications of speech synthesized speech can be done through ESNOLA.

Main advantage of this method is that, it allows selection of smaller parts of phonemes called partnemes as concatenation units that reduces the size of the speech inventory. ESNOLA supports the introduction of jitter shimmer and complexity perturbations that leads to naturalness in phonetic quality of synthesized speech.

2.2.2 Corpus based speech synthesis

This method uses data driven approach. This method depends on the availability of good speech inventory, having good phonetic and prosodic features for the language under consideration.

Segmentation and labelling of the speech inventory is major issue related to corpus based approach that can be achieved using automatic segmentation algorithm.

2.2.2.1 Unit Selection Synthesis:

In this method multiple instance of speech units having different prosodic features are stored. Here, unit is selected from the database based on two costs: A target cost and a concatenation cost.

Target Cost:

It estimates how similar the features of database speech unit are to the features of the desired speech unit..It comprises of target sub-costs where each target sub cost is a cost of a single attribute of a speech unit such as energy, pitch etc.

The target cost can be calculated as: $Ct(ti,vi) = \sum_{pj=1} wtj\ Ctj(ti,vi)$

Where, ti is the target unit , Vi is the candidate unit and p is the number of sub-costs used. Ctj is the jth target sub-cost, Wtj it is the weight given to the jth target sub-cost.

Concatenation Cost:

It is a measure of how well two speech units join and match each other when they are concatenated . The concatenation cost also comprises of multiple subcosts where each of these sub-costs is related to a specific continuity metric such as spectral continuity etc.

The concatenation cost can be calculated as: $Cc(vi-1,vi) = \sum_{qj=1} wcj\ Ccj(vi-1,vi)$

Where vi-1 and vi are candidate speech units for the (i-1)th and ith target speech units, q is the total number of subcosts used and Wcj is the weight associated with the subcost Ccj .

An exhaustive search is performed to select optimum speech units from the speech database.

This method is sample based method and main advantage of this system is that it requires minimum or no signal processing and it gives Good quality in naturalness of synthesized speech. But Requires the main drawback of this system is that it requires a large speech database.

2.2.2.2 Statistical Parametric Synthesis:

Statistical parametric synthesis makes use of averaged acoustic inventories that are extracted from the speech corpus. The extracted parameters of speech are the spectral

parameters such are cepstral coefficients or line spectral pairs, and excitation parameters such as fundamental frequency.

Statistical Parametric synthesis has the advantages of requiring less memory to store the parameters of the model, rather than the data itself and it allows more Variation in the speech produced for example, an original voice can be converted into another voice. The most commonly used statistical parametric speech synthesis technique is the Hidden Markov Model (HMM) synthesis [4].

2.2.2.3 Hidden Markov Model (HMM) synthesis:

HMM synthesis has two phases: A training phase and a synthesis phase.

In the training phase speech parameters are extracted from utterances in the speech training database and they are modeled as HMMs and In the synthesis phase the words to be synthesized their corresponding HMMs are identified from the database and parameters are extracted from these HMMS. Finally speech is synthesized from these extracted parameters.

This method is based on parameters. Main advantage of HMM based parametric speech synthesis is the flexibility since speech is stored in the form of parameters and it is easy to modify these parameters. Apart from it, it requires small speech inventory but it has the disadvantage of poor quality in the naturalness of the synthesized speech due to over smoothing of the parameters in the statistical model and required more Signal processing [3].

2.2.3 Hybrid Text To Speech Synthesis:

The Hybrid TTS approach is a combination of the two main approaches of synthesis: Concatenative synthesis and Statistical Synthesis. The hybrid TTS combines the characteristics of smooth transitions between adjacent speech segments of a Statistical TTS with the naturalness of a Concatenative TTS. This is achieved by interweaving natural speech segments and statistically generated speech segments. The statistical segments are positioned so as to smooth discontinuities in the synthesized speech, while enabling as far as possible natural speech sequences as they appear in the training inventory disadvantages of this system are the Degradation in speech quality when CTTS speech inventory is small and more signal processing requirement [5].

2.3 Articulatory Synthesis

Articulatory synthesis refers to computational techniques for synthesizing speech based on models of the human vocal tract and the articulation processes occurring there. “Articulatory speech synthesis models the natural speech production process as accurately as possible. This is accomplished by creating a synthetic model of human physiology and making it speak. [6].Articulatory synthesis systems comprise (i) a module for the generation of vocal tract movements (control model), (ii) a module for converting this movement information into a continuous succession of vocal tract geometries (vocal tract model), and (iii) a module for the generation of acoustic signals on the basis of this articulatory information (acoustic model).[7]

2.3.1 Vocal Tract Models:

The task of vocal tract models is to generate the complete geometrical information concerning the vocal tract (shape and position of all vocal tract organs, i.e. lips, tongue, palate, velum, pharynx, larynx, nasal cavity) and its variation over time. Shape, position, and motion of movable vocal tract organs are generated on the basis of the time functions of all vocal tract parameters defined by the model [7]. A typical set of vocal tract parameters are: position of jaw, upper lips, lower lips, tongue tip, tongue body, velum, and larynx [6]. Vocal tract models can be subdivided into statistical, biomechanical, and geometrical models [7].

Statistical models are based on large corpora of vocal tract movements measured by different techniques (MRI, EMA, or X-Ray [7]. Biomechanical models aim to model the physiological basis of all vocal tract organs and their neuromuscular control [7]. For geometrical models the positioning and shape of the vocal tract organs is calculated by using a set of a priori defined vocal tract parameters.

2.3.2 Acoustic Models:

The task of the acoustic models is to calculate the time varying air flow and air pressure distribution within the vocal tract and to calculate the acoustic speech signal radiated from the facial region of the model[8]. The input information for acoustic models is lung pressure, subglottal air flow, and the geometric shape of the vocal tract tube (trachea, glottis, pharyngeal, oral, and nasal tract) for each time instant. A timevarying tube model is specified from the geometrical vocal tract model information, which represents the vocal tract cavities (trachea, pharynx, nasal, and oral cavity) [7]. Acoustic models can be subdivided into reflection type line analog models, transmission line circuit analog models, hybrid time-frequency domain models, and finite element wave propagation models

In the case of reflection type line analog models forward and backward traveling partial flow or pressure waves are calculated for each vocal tract tube section in the time domain on the basis of scattering equations which reflect the impedance discontinuity at tube junctions [12]. In the case of transmission line circuit analog models, pressure and flow within each vocal tract tube section is calculated by a digital simulation of electrical circuit elements, representing the acoustic and aerodynamic properties within each vocal tract tube section.

2.3.3 Glottis Models:

The task of glottis models is to generate the acoustic source signal for phonation and its insertion into the vocal tract tube model[7]. The source signal is propagated through the supraglottal cavities (pharyngeal, oral and nasal cavity) as well as through the subglottal cavities (trachea, lungs) by the acoustic model. Glottis models can be subdivided into self-oscillating models, parametric glottal area models, and parametric glottal flow models [11].

2.3.4 Noise Source Models:

The task of noise source models is to generate and to insert noise source signals into the acoustic transmission line model. Noise signals result from turbulent air flow, mainly occurring downstream in front of a vocal tract constriction in the case of a high value of volume flow. Noise source models can be subdivided into parametric and generic noise source models [9][10].

3. CONCLUSION

Speech synthesis has been developed steadily over the last decades and it has been incorporated into several new applications. Text to speech synthesis is a rapidly growing aspect of computer technology and is increasingly playing a more important role in the way we interact with the system and interfaces across a variety of platforms. We have identified the various operations and processes involved in text to speech synthesis.

For most applications, the intelligibility and comprehensibility of synthetic speech have reached the acceptable level. However, in prosodic, text preprocessing, and pronunciation fields there is still much work and improvements to be done to achieve more natural sounding speech. The three basic methods used in speech synthesis have been introduced in Chapter 2. The most commonly used techniques in present systems are based on formant and Concatenative synthesis. The latter one is becoming more and more popular since the methods to minimize the problems with the discontinuity effects in concatenation points are becoming more effective. The Concatenative method provides more natural and individual sounding speech, but the quality with some consonants may vary considerably and the controlling of pitch and duration may be in some cases difficult, especially with longer units. With concatenation methods the collecting and labeling of speech samples have usually been difficult and very time-consuming. With formant synthesis the quality of synthetic speech is more constant, but the speech sounds slightly more unnatural and individual sounding speech is more difficult to achieve. Formant synthesis is also more flexible and allows a good control of fundamental frequency [1]. The third basic method, the Articulatory synthesis, is perhaps the most feasible in theory especially for stop consonants because it models the human articulation system directly. On the one hand, the Articulatory based methods are usually rather complex and the computational load is high [7][8].

4. REFERENCES

[1] Sami Lemmetty. Review of Speech Synthesis Technology. Helsinki University of Technology Department of Electrical and Communications Engineering. March 30, 1999.

[2] Rubeena A. Khan , J. S. Chitode, Concatenative Speech Synthesis: A Review, *International Journal of Computer Applications (0975 – 8887). Volume 136 – No.3, February 2016.pg-1 to 4.*

[3] Raitio, Tuomo, et al. "HMM-based speech synthesis utilizing glottal inverse filtering." Audio, Speech, and Language Processing, IEEE Transactions on vol.19, no.1, 2011, pp. 153-165.

[4] Heiga Zen, Keiichi Tokuda, Alan W. Black ,"Statistical parametric speech synthesis", Speech Communication vol.51,no.11,2009,pp. 1039–1064.

[5] Pertti Palo. A Review of Articulatory Speech Synthesis. Espoo, June 5, 2006

[6] Birkholz P, Martin L, Willmes K, Kröger BJ, Neuschaefer-Rube C (2015) The contribution of phonation type to the perception of vocal emotions in German: An articulatory synthesis study. Journal of the Acoustical Society of America 137:1503-1512

[7] Louis Goldstein and Carol A. Fowler. Articulatory Phonology: A phonology for public language use

[8] Richard S, Mc gowan and Alice Faber. Introduction to papers on speech recognition and perception from an articulatory point of view.

[9] Shuangyu Chang. A Syllable, Articulatory-F eature, and Stress-Accent Model of Speech Recognition. September 2002

[10] Kelly and Lochbaum 1962, Liljencrants 1985, Meyer et al. 1989, Kröger 1998.(e.g. Flanagan 1975, Maeda 1982, Birkholz et al. 2007.

International Journal of Computer Applications Technology and Research
Volume 6–Issue 12, 516-518, 2017, ISSN:-2319–8656

Integration of Database Management System with Cloud

Sushma V
GSSSIETW,
Mysuru, India

Abstract: In recent years, outsourcing the database has become an important component of cloud computing. Due to the continuous advancements in a network technology, the cost required to transmit a terabyte of data over long distances as decreased significantly in the past decade. In addition, the total cost of data management is five to ten times higher than the initial acquisition cost. As a result, there is a growing interest in outsourcing database management tasks to third parties that can provide these tasks for much lower cost due to the economy of scale. This new outsourcing model has the benefits of reducing the cost for running Database Management System (DBMS) independently.

Keywords: SQL (Structured Query Language), RDBMS (Relational Database Management System), BigData, IaaS(Infrastructure as a service), PaaS(Platform as a service component)

1. INTRODUCTION

A Cloud database management system is a distributed database that delivers computing as a service instead of a product. It is the sharing of resources, software, and information between multiple devices over a network which is mostly the internet. It is expected that this number will grow significantly in the future. As a result, there is a growing interest in outsourcing database management tasks to third parties that can provide these tasks for much lower cost due to the economy of scale just like putting it into the cloud. In this paper, we discuss the recent trend in database management system and the possibilities of making it as one of the services offered in the cloud.

2. THE TRADITIONAL WAY

Storing of data on computer hard disk and secondary storage is too main stream and has been facing many problems these days, due to data loss, failure and difficulty in remote access by concurrent users. To avoid data loss and allow concurrent users access, Cloud Computing has been put into use which allows data to be stored on a virtual cloud like storage area, which can be accessed by a user from remote location and data is even safe. Using Database as a Cloud Service allows the user to access the data stored on an online database (present on a cloud) and is hence proving to be an efficient technological advancement these days.

3. DBMS AS A CLOUD SERVICE

Most DBMS or database management systems are simply software packages that users can acquire to create, maintain or use a database. However, since the introduction of cloud computing, DBMS has morphed into an entirely new type of service with its own unique benefits and task specific advantages. For one thing, any type of cloud service model will have to employ a dedicated cloud DBMS in order to truly provide customers with excellent access to data and databases. Traditional DBMS's are simply not set up or equipped to deal with the demands of cloud computing. And of course, if DBMS was deployed as a service as part of a larger package provided, it would likely be much more efficient in its duties and therefore cheaper in the long run.

The concept of the DBMS has been around since the beginning of commercial computing; such as the navigational DBMS of the1960's. Database management systems are one of the oldest integral components of computing, essentially making it possible to scan, retrieve and organize data on hard drives and networks. All DBMS, despite whether traditional or cloud-based, are essentially communicators that function as middlemen between the operating system and the database.

How is a cloud DBMS different a traditional one? For one thing, cloud-based DBMS are extremely scalable. They are able to handle volumes of data and processes that would exhaust a typical DBMS. Despite their scalability however, cloud DBMS are still somewhat lacking in their ability to scale up to extremely large processes; this is expected to be remedied in the coming months and years however. Currently, the use of cloud DBMS's are principally used in the testing and development of new cloud applications and processes[1]. But while a stand-alone DBMS can be used on a cloud infrastructure; most are not designed to take full advantage of cloud resources. DBMS as a cloud service-type models seek to capitalize on the disparity between antiquated DBMS models and their lack of full cloud functionality.

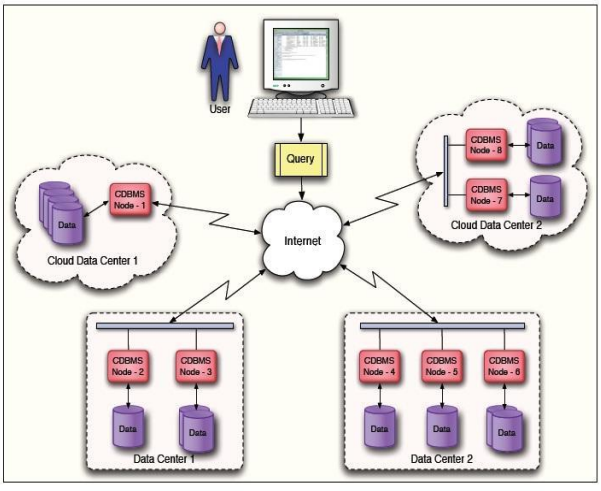


Figure. 1 Basic model of Cloud Database

4. CLOUD DATABASE STRATEGY

One of the most significant (and complex) cloud infrastructure issues facing cloud providers of many types is deciding how database support will be offered in the cloud, which is also leading to selling Database as a Service[2]. The wrong cloud

database strategy can create application performance problems significant enough to discredit a cloud service, forcing the provider to incur additional costs to establish credibility with users.

The cloud database issue is complicated because it sits at the intersection of two cloud infrastructure models, two storage service models and two database management systems (DBMS) models. Sorting out the details will require cloud to consider their infrastructure, network performance and service goals.

The following services models can affect cloud database support:

4.1 Single and multi-site cloud infrastructure models

The two cloud infrastructure models differ in the way that resources are allocated to customers. In the single-site model, a customer's applications run within a single data centre in the cloud, even if multiple data centres are available. This means that the storage and/or DBMS resources used by a customer can be contained within a single storage area network (SAN), and that the customer's application performance in the cloud can likely match that of a standard data centre that uses virtualization. In the multi-site model, the customer's applications can draw on resources from multiple data centres, which mean that making the connection between the application and the database resources could involve WAN connectivity that limits performance[3]. Whichever choice they make, service providers must be ready to address the issues that come with single- or multi-site cloud infrastructure.

4.2 Storage and database service models

The storage service models available to a cloud planner are Storage as a Service or the more complex Database as a Service. With storage services, the customer will access virtual storage devices as though they were native disk arrays, which means that the applications will send storage protocols (such as Fibre Channel over Ethernet or IP SCSI) over any network connection. In the relatively new Database as Service offerings, applications will access storage through a cloud DBMS that will accept high-level database commands and return the required results. This can create a less delay-sensitive connection, so it is better suited to cloud configurations where storage might be distributed over multiple sites.

Another major cloud database planning decision is whether a cloud database service should be based on the popular relational database management system (RDBMS) and its Structured Query Language (SQL) standards, based on a lighter-weight RDBMS without SQL, or based on a non-relational structure like the 'Google BigTable' structure that gives users dynamic control over data layout and format

5. DBMS IN CLOUD ARCHITECTURE

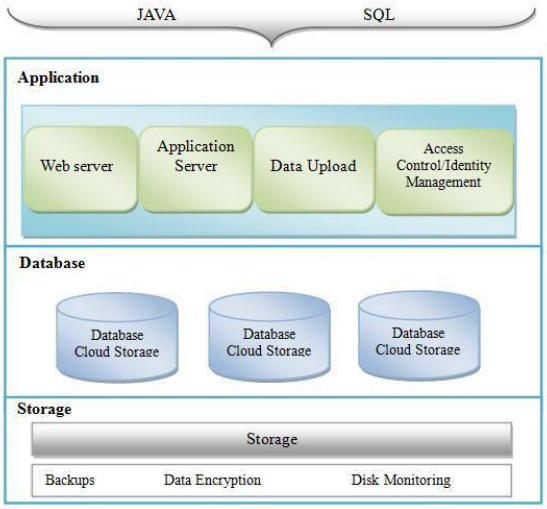


Figure 2 DBMS in Cloud Architecture

Above is a proposed DBMS in Cloud Architecture, first layer is the storage, followed by databases and the upper layer is application layer. In terms of performance, it provides efficient data access with a better distribution of values for some data. Stores frequently used SQL statements in memory, avoiding the need for time-consuming recompilation at run-time[4]. Produces a detailed report on each step used for data access, allowing you to accurately implement performance enhancements. Data is encrypted when stored or backed up, without any need for programming to encrypt and decrypt

6. THE VALUE OF DBMS AS A CLOUD SERVICE

6.1 Advantages

1) Database as a Service has advantages beyond marketing. With a cloud DBMS, storage virtualization isn't directly visible to the applications, which gives operators more latitude in how they manage storage resources.

2) With a direct storage model, a mechanism for storage virtualization that protects customers' data from being accessed by others but still makes the virtual disks look "real" is essential
3) It helps in controlling the performance of applications that use storage extensively

4) It's easy to provide cloud database services as part of a cloud Platform as a Service (PaaS) offering, but the

5) applications may have to be written to access cloud database services in some Infrastructure as a Service (IaaS) configurations[5].
6) 5) Offering Database as a Service can help by replacing storage input/output (I/O) with simply sending a query and a return of results.

6.2 Demerits

1) If customers are likely to access the storage/DBMS across multiple data centres in cloud, the performance implications may be critical.

2) Privacy has been the most important issue when it comes to cloud computing. The companies cannot afford to leak out information that is stored in their database. If there is encryption of data in the database, then it is quite easy to store it in an easy way.

3) Query and Transactional workload: Transactional workload determines the overall time that will be required[6]. But we cannot determine the Query workload, since there can be multiple users firing numerous queries at same time, which would be time consuming.

4) Internet Speed: The speed of data transfer in the data centre is comparatively very high as compared to the speed of internet that is used to access the data centre. This is a barrier to the performance of cloud database[7].

7. CONCLUSION

Database Management Systems as a cloud service are engineered to run as a scalable, elastic service available on a cloud infrastructure. Cloud DBMSs will have an impact for vendors desiring a less expensive platform for development. In this paper, we presented the idea of DBMS in the cloud, the possibilities to be offered as one of the services offered by promising capability of cloud computing, that is to be a DBMS as a Service. In this paper we proposed architecture of DBMS in the cloud.

8. REFERENCES

[1] A. Thusoo, J. S. Sarma, N. Jain, Z. Shao, P. Chakka, S. Anthony, H. Liu, P. Wyckoff, and R. Murthy, "Hive: a warehousing solution over a map-reduce framework," Proc. VLDB Endow., vol. 2, pp. 1626–1629, August 2009.

[2] M. Stonebraker, S. Madden, D. J. Abadi, S. Harizopoulos, N. Hachem, and P. Helland. The end of an architectural era: (it's time for a complete rewrite). In VLDB, pages 1150–1160, 2007

[3] J. Lee, M. Muehle, N. May, F. Faerber, V. S. H. Plattner, J. Krueger, and M. Grund. High-performance transaction processing in SAP HANA. IEEE Data Eng. Bull., 36(2):28–33, 2013.

[4] GravelleR, "Should You Move Your MySQL Database to the Cloud?", http://www.databasejournal.com/ features/mssql/should-you-move-your-mysql-database-tothe-cloud.html.

[5] A. Williamson, "Comparing cloud computing providers," Cloud Comp. J., vol. 2, no. 3, pp. 3–5, 2009.

[6] B. R. Kandukuri, R. Paturi V, A. Rakshit, "Cloud Security Issues", In Proceedings of IEEE International Conference on Services Computing, pp. 517-520, 2009.

[7] V. Krishna Reddy, B. Thirumal Rao, Dr. L.S.S. Reddy, P.Sai Kiran "Research Issues in Cloud Computing " Global Journal of Computer Science and Technology, Volume 11, Issue 11, July 2011.